

Cisco APIC Layer 2 Networking Configuration Guide, Release 6.0(x)

First Published: 2022-06-17

Last Modified: 2024-05-21

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

© 2022 –2024 Cisco Systems, Inc. All rights reserved.



Trademarks

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS REFERENCED IN THIS DOCUMENTATION ARE SUBJECT TO CHANGE WITHOUT NOTICE. EXCEPT AS MAY OTHERWISE BE AGREED BY CISCO IN WRITING, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENTATION ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED.

The Cisco End User License Agreement and any supplemental license terms govern your use of any Cisco software, including this product documentation, and are located at: https://www.cisco.com/c/en/us/about/legal/cloud-and-software/software-terms.html. Cisco product warranty information is available at https://www.cisco.com/c/en/us/products/warranty-listing.html. US Federal Communications Commission Notices are found here https://www.cisco.com/c/en/us/products/us-fcc-notice.html.

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 products and features described herein as in development or available at a future date remain in varying stages of development and will be offered on a when-and if-available basis. Any such product or feature roadmaps are subject to change at the sole discretion of Cisco and Cisco will have no liability for delay in the delivery or failure to deliver any products or feature roadmap items that may be set forth in this document.

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.

The documentation set for this product strives to use bias-free language. For the purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on RFP documentation, or language that is used by a referenced third-party product.

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)

Trademarks



CONTENTS

PREFACE

Trademarks iii

CHAPTER 1

New and Changed 1

New and Changed Information 1

CHAPTER 2

Cisco ACI Forwarding 3

ACI Fabric Optimizes Modern Data Center Traffic Flows 3

VXLAN in ACI 4

Layer 3 VNIDs Facilitate Transporting Inter-subnet Tenant Traffic 6

Transmission of Spanning Tree Protocol BPDU 8

CHAPTER 3

Prerequisites for Configuring Layer 2 Networks 9

Layer 2 Prerequisites 9

CHAPTER 4

Networking Domains 11

Networking Domains 11

Related Documents 12

Bridge Domains 12

About Bridge Domains 12

VMM Domains 12

Virtual Machine Manager Domain Main Components 12

Virtual Machine Manager Domains 13

Configuring Physical Domains 13

Configuring a Physical Domain 13

CHAPTER 5

Bridging 15

Bridged Interface to an External Router 15
Bridge Domains and Subnets 16
Bridge Domain Options 19
Creating a Tenant, VRF, and Bridge Domain Using the GUI 21
Creating a Tenant, VRF, and Bridge Domain Using the NX-OS Style CLI 23
Configuring an Enforced Bridge Domain 24
Configuring an Enforced Bridge Domain Using the NX-OS Style CLI 25
Configuring Flood in Encapsulation for All Protocols and Proxy ARP Across Encapsulations 26
Configuring Flood in Encapsulation 31
Configuring Flood in Encapsulation Using the Cisco APIC GUI 32
Configuring Flood in Encapsulation Using the NX-OS Style CLI 32

CHAPTER 6 EPGs 35
About Endpoint Groups 35
Endpoint Groups 35
ACI Policy Configuration in EPG Shutdown 37
Access Policies Automate Assigning VLANs to EPGs 37
Per Port VLAN 38
VLAN Guidelines for EPGs Deployed on vPCs 40
Deploying an EPG on a Specific Port 41
Deploying an EPG on a Specific Node or Port Using the GUI 41
Deploying an EPG on a Specific Port with APIC Using the NX-OS Style CLI 42
Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port 44
Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port 44
Creating Domains, and VLANs to Deploy an EPG on a Specific Port Using the GUI 44
Creating AEP, Domains, and VLANs to Deploy an EPG on a Specific Port Using the NX-OS Style CLI 46
Validating Overlapping VLANs 47
Validating Overlapping VLANs Using the GUI 47
Deploying EPGs to Multiple Interfaces Through Attached Entity Profiles 47
Deploying an Application EPG through an AEP or Interface Policy Group to Multiple Ports 47
Deploying an EPG through an AEP to Multiple Interfaces Using the APIC GUI 48
Deploying an EPG through an Interface Policy Group to Multiple Interfaces Using the NX-OS Style CLI 49

Intra-EPG Isolation 50
Intra-EPG Endpoint Isolation 50
Intra-EPG Isolation for Bare Metal Servers 50
Intra-EPG Isolation for Bare Metal Servers 50
Configuring Intra-EPG Isolation for Bare Metal Servers Using the GUI 51
Configuring Intra-EPG Isolation for Bare Metal Servers Using the NX-OS Style CLI 52
Intra-EPG Isolation for VMWare vDS 54
Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch 54
Configuring Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch using the GUI 57
Configuring Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch using the NX-OS Style CLI 58
Configuring Intra-EPG Isolation for Cisco ACI Virtual Edge 60
Intra-EPG Isolation Enforcement for Cisco ACI Virtual Edge 60
Configure Intra-EPG Isolation for Cisco ACI Virtual Edge Using the GUI 60
Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab 61
View Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab 62
Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Virtual Networking Tab 63
View Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Virtual Networking Tab 63
Configure Intra-EPG Isolation for Cisco ACI Virtual Edge Using the NX-OS Style CLI 64
Troubleshooting 64
Troubleshooting Endpoint Connectivity 64
Inspecting the Endpoint Status 65
Inspecting the Tunnel Interface Status 66
Performing a Traceroute Between Endpoints 66
Configuring Atomic Counters 67
Configuring a Tenant SPAN Session Using the Cisco APIC GUI 68
Verifying IP-Based EPG Configurations 68
Verifying IP-Based EPG Configurations Using the GUI 68
Verifying IP-EPG Configurations Using Switch Commands 69

CHAPTER 7 **Access Interfaces** 71
About Access Interfaces 71

Physical Ports Configuration 74
Configuring Leaf Switch Physical Ports Using the Interface Configuration Model Using the GUI 74
Migrating Interfaces From the Selector and Profile to Interface Configuration Using the GUI 76
Modifying the Interface Configuration Using the GUI 77
Viewing the Interface Configuration Using the GUI 77
Configuring Physical Ports in Leaf Nodes and FEX Devices Using the NX-OS CLI 78
Port Channels 81
PC Host Load Balancing Algorithms 81
Cisco ACI Leaf Switch Port Channel Configuration Using the GUI 82
Configuring Port Channels in Leaf Nodes and FEX Devices Using the NX-OS CLI 84
Virtual Port Channels in Cisco ACI 89
Cisco ACI Virtual Port Channel Workflow 91
Defining a vPC Using the GUI 92
Configuring Virtual Port Channels in Leaf Nodes and FEX Devices Using Profiles and Selectors 93
vPC With the Same Leaf Switch Interfaces Across Two Leaf Switches With Combined Profiles 93
vPC With the Same Leaf Switch Interfaces Across Two Leaf Switches with Individual Profiles 95
Configuring a Cisco ACI Leaf Switch Virtual Port Channel Using the Interface Configuration Model Using the GUI 97
Configuring Virtual Port Channels in Leaf Nodes and FEX Devices Using the NX-OS CLI 98
Reflective Relay (802.1Qbg) 103
Enabling Reflective Relay Using the GUI 103
Enabling Reflective Relay Using the NX-OS CLI 104
Configuring Port, PC, and vPC Connections to FEX Devices 105
ACI FEX Guidelines 106
FEX Virtual Port Channels 106
Configuring a FEX Connection to the ACI Leaf Switch Using the GUI 108
Configuring FEX Connectivity to an ACI leaf switch Using Profiles with the NX-OS-Style CLI 109
Configuring Port Profiles 110
Port Profile Configuration Summary 115
Changing an Uplink to a Downlink or Downlink to an Uplink Using the GUI 120
Changing an Uplink to a Downlink or Downlink to an Uplink Using the NX-OS-Style CLI 120
Verifying Port Profile Configuration and Conversion Using the NX-OS Style CLI 121
Editing an Interface Configuration 122

CHAPTER 8 FCoE Connections 123

Supporting Fibre Channel over Ethernet Traffic on the Cisco ACI Fabric 123
Fibre Channel over Ethernet Guidelines and Limitations 125
Fibre Channel over Ethernet Supported Hardware 125
Configuring FCoE Using the APIC GUI 126
FCoE GUI Configuration 126
FCoE Policy, Profile, and Domain Configurations 126
Deploying FCoE vFC Ports Using the APIC GUI 129
Deploying EPG Access to vFC Ports Using the APIC GUI 135
Deploying the EPG to Support the FCoE Initiation Protocol 138
Undeploying FCoE Connectivity Using the APIC GUI 140
Configuring FCoE Using the NX_OS Style CLI 142
FCoE NX-OS Style CLI Configuration 142
Configuring FCoE Connectivity Without Policies or Profiles Using the NX-OS Style CLI 142
Configuring FCoE Connectivity With Policies and Profiles Using the NX-OS Style CLI 145
Configuring FCoE Over FEX Using NX-OS Style CLI 148
Verifying FCoE Configuration Using the NX-OS Style CLI 150
Undeploying FCoE Elements Using the NX-OS Style CLI 150
SAN Boot with vPC 152
Configuring SAN Boot with vPC Using the GUI 153
SAN Boot with vPC Configuration Using the CLI 156

CHAPTER 9 Fibre Channel NPV 157

Fibre Channel Connectivity Overview 157
NPV Traffic Management 159
Automatic Uplink Selection 159
Traffic Maps 160
Disruptive Auto Load Balancing of Server Logins across NP Links 160
FC NPV Traffic Management Guidelines 160
SAN A/B Separation 161
SAN Port Channels 162
Fibre Channel N-Port Virtualization Guidelines and Limitations 163
Fibre Channel N-Port Virtualization Supported Hardware 164

Fibre Channel N-Port Virtualization Interoperability 164
Fibre Channel NPV GUI Configuration 165
Configuring a Native Fibre Channel Port Profile Using the GUI 165
Configuring a Native FC Port Channel Profile Using the GUI 167
Deploying Fibre Channel Ports 168
Configuring a Traffic Map for a Fibre Channel Port 170
Fibre Channel NPV NX-OS-Style CLI Configuration 171
Configuring Fibre Channel Interfaces Using the CLI 171
Configuring Fibre Channel NPV Policies Using the CLI 173
Configuring an NPV Traffic Map Using the CLI 174
Fibre Channel NPV REST API Configuration 175
Configuring FC Connectivity Using the REST API 175

CHAPTER 10 802.1Q Tunnels 179
About ACI 802.1Q Tunnels 179
Configuring 802.1Q Tunnels Using the GUI 181
Configuring 802.1Q Tunnel Interfaces Using the APIC GUI 181
Configuring 802.1Q Tunnels Using the NX-OS Style CLI 183
Configuring 802.1Q Tunnels Using the NX-OS Style CLI 183
Example: Configuring an 802.1Q Tunnel Using Ports with the NX-OS Style CLI 185
Example: Configuring an 802.1Q Tunnel Using Port-Channels with the NX-OS Style CLI 185
Example: Configuring an 802.1Q Tunnel Using Virtual Port-Channels with the NX-OS Style CLI 186

CHAPTER 11 Q-in-Q Encapsulation Mapping for EPGs 187
Q-in-Q Encapsulation Mapping for EPGs 187
Configuring Q-in-Q Encapsulation Mapping for EPGs Using the GUI 188
Enabling Q-in-Q Encapsulation on Specific Leaf Switch Interfaces Using the GUI 188
Enabling Q-in-Q Encapsulation for Leaf Interfaces With Fabric Interface Policies Using the GUI 189
Mapping an EPG to a Q-in-Q Encapsulation-Enabled Interface Using the GUI 190
Mapping EPGs to Q-in-Q Encapsulated Leaf Interfaces Using the NX-OS Style CLI 191

CHAPTER 12 Breakout Ports 193
Configuration of Breakout Ports 193
Guidelines and Limitations for Dynamic Breakout Ports on Down Links 194

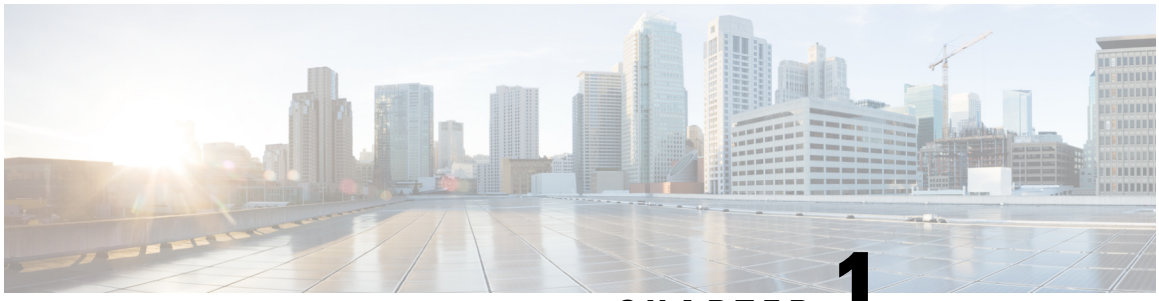
Guidelines and Limitations for Auto Breakout Ports on Fabric Links 198
Configuring Breakout Ports With Profiles and Selectors Using the GUI 200
Configuring Breakout Ports With Profiles and Selectors Using the GUI 203
Configuring Breakout Ports With the Interface Configuration Using the GUI 206
Configuring Dynamic Breakout Ports Using the NX-OS Style CLI 206

CHAPTER 13 **Proxy ARP 211**
About Proxy ARP 211
Guidelines and Limitations 217
Proxy ARP Supported Combinations 217
Configure Proxy ARP Using the Advanced GUI 218
Configure Proxy ARP Using the Cisco NX-OS Style CLI 218

CHAPTER 14 **Traffic Storm Control 221**
About Traffic Storm Control 221
Storm Control Guidelines and Limitations 221
Configuring a Traffic Storm Control Policy Using the GUI 224
Configuring a Traffic Storm Control Policy Using the NX-OS Style CLI 225
Configuring a Storm Control SNMP Trap 227
Storm Trap 227

CHAPTER 15 **MACsec 229**
About MACsec 229
Guidelines and Limitations for MACsec on switches 230
Configuring MACsec for Fabric Links Using the GUI 234
Configuring MACsec for Access Links Using the GUI 234
Configuring MACsec Parameters Using the APIC GUI 235
Configuring MACsec Keychain Policy Using the GUI 235
Configuring MACsec Using the NX-OS Style CLI 236

CHAPTER 16 **Fabric Port Tracking 239**
About Fabric Port Tracking 239
Configuring Fabric Port Tracking Using the GUI 240



CHAPTER 1

New and Changed

This chapter contains the following section:

- New and Changed Information, on page 1

New and Changed Information

The following table provides an overview of the significant changes to this guide for this release. The table does not provide an exhaustive list of all changes made to the guide or of the new features up to this release.

Table 1: New Features and Changed Information for Cisco APIC 6.0(3)

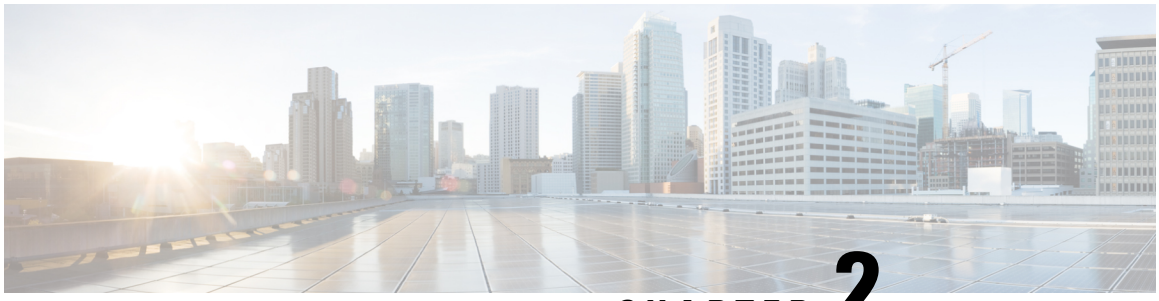
Feature	Description	Where Documented
100Gb to 25Gb dynamic breakout support for GX2 switches	The Cisco N9K-C9348D-GX2A, N9K-C9364D-GX2A, and N9K-C9332D-GX2B switches supports 100Gb to 25Gb dynamic breakouts.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194

Table 2: New Features and Changed Information for Cisco APIC 6.0(2)

Feature	Description	Where Documented
QDD-400G-SR4-BD optic support on 400Gb ports	The QDD-400G-SR4-BD optic is supported on 400Gb ports. The peer node with the 100Gb speed must use QSFP-100G-SR1.2 optics. An MPO-to-breakout splitter cable with QDD-400G-SR4-BD and 4 x QSFP-100G-SR1.2 on the ends is supported.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194 Guidelines and Limitations for Auto Breakout Ports on Fabric Links, on page 198
Dynamic breakout support for Cisco N9K-C9400-SW-GX2A switches with the Cisco N9K-X9400-16W linecard	The Cisco N9K-C9400-SW-GX2A switch with the Cisco N9K-X9400-16W linecard supports dynamic breakouts (both 100Gb and 40Gb) on profiled QSFP ports.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194

Table 3: New Features and Changed Information for Cisco APIC 6.0(1)

Feature	Description	Where Documented
REST API procedures	The REST API procedures are now provided at developer.cisco.com for the 6.0(1) release and all later releases.	https://developer.cisco.com/docs/apic-rest-api-configuration-guide/
4x100G breakout support on N9K-C9348D-GX2A, N9K-C9364D-GX2A, and N9K-C9332D-GX2B	4x100G breakout is supported on the Cisco N9K-C9348D-GX2A, N9K-C9364D-GX2A, and N9K-C9332D-GX2B switches' access ports and fabric ports.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194 Guidelines and Limitations for Auto Breakout Ports on Fabric Links, on page 198
400G access port support for QDD-4X100G-FR-S, QDD-4X100G-LR-S, and QDD-400G-DR4-S	QDD-4X100G-FR-S, QDD-4X100G-LR-S, and QDD-400G-DR4-S optics are supported on 400Gb access ports.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194
400G fabric port support for QDD-4X100G-FR-S and QDD-4X100G-LR-S	QDD-4X100G-FR-S and QDD-4X100G-LR-S optics are supported on 400Gb fabric ports.	Guidelines and Limitations for Auto Breakout Ports on Fabric Links, on page 198
100Gb peer node support for QSFP-100G-LR-S	QSFP-100G-LR-S optics are supported with the peer node with the 100Gb speed for access ports and fabric ports.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194 Guidelines and Limitations for Auto Breakout Ports on Fabric Links, on page 198
MPO to breakout splitter cable support for QDD-4X100G-LR-S	QDD-4X100G-LR-S optics are supported with the MPO to breakout splitter cable for access ports and fabric ports.	Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194 Guidelines and Limitations for Auto Breakout Ports on Fabric Links, on page 198



CHAPTER 2

Cisco ACI Forwarding

This chapter contains the following sections:

- ACI Fabric Optimizes Modern Data Center Traffic Flows, on page 3
- VXLAN in ACI, on page 4
- Layer 3 VNIDs Facilitate Transporting Inter-subnet Tenant Traffic, on page 6
- Transmission of Spanning Tree Protocol BPDU, on page 8

ACI Fabric Optimizes Modern Data Center Traffic Flows

The Cisco ACI architecture addresses the limitations of traditional data center design, and provides support for the increased east-west traffic demands of modern data centers.

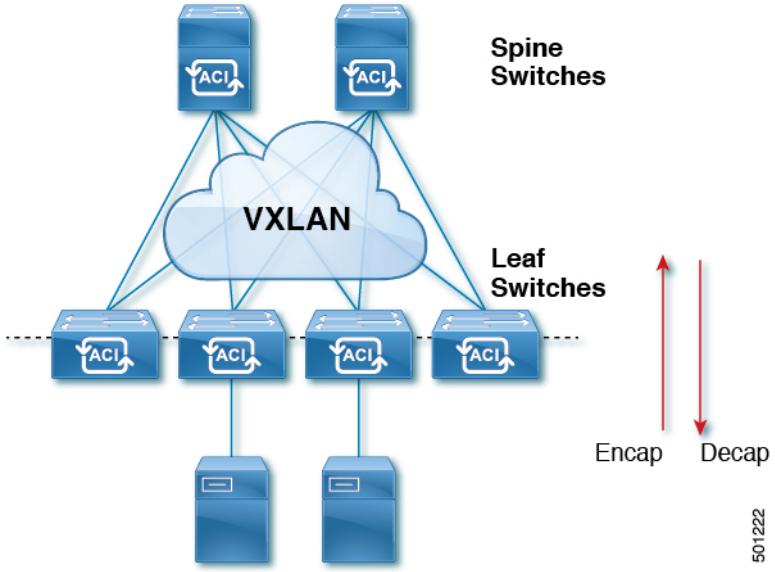
Today, application design drives east-west traffic from server to server through the data center access layer. Applications driving this shift include big data distributed processing designs like Hadoop, live virtual machine or workload migration as with VMware vMotion, server clustering, and multi-tier applications.

North-south traffic drives traditional data center design with core, aggregation, and access layers, or collapsed core and access layers. Client data comes in from the WAN or Internet, a server processes it, and then it exits the data center, which permits data center hardware oversubscription due to WAN or Internet bandwidth constraints. However, Spanning Tree Protocol is required to block loops. This limits available bandwidth due to blocked links, and potentially forces traffic to take a suboptimal path.

In traditional data center designs, IEEE 802.1Q VLANs provide logical segmentation of Layer 2 boundaries or broadcast domains. However, VLAN use of network links is inefficient, requirements for device placements in the data center network can be rigid, and the VLAN maximum of 4094 VLANs can be a limitation. As IT departments and cloud providers build large multi-tenant data centers, VLAN limitations become problematic.

A spine-leaf architecture addresses these limitations. The ACI fabric appears as a single switch to the outside world, capable of bridging and routing. Moving Layer 3 routing to the access layer would limit the Layer 2 reachability that modern applications require. Applications like virtual machine workload mobility and some clustering software require Layer 2 adjacency between source and destination servers. By routing at the access layer, only servers connected to the same access switch with the same VLANs trunked down would be Layer 2-adjacent. In ACI, VXLAN solves this dilemma by decoupling Layer 2 domains from the underlying Layer 3 network infrastructure.

Figure 1: ACI Fabric



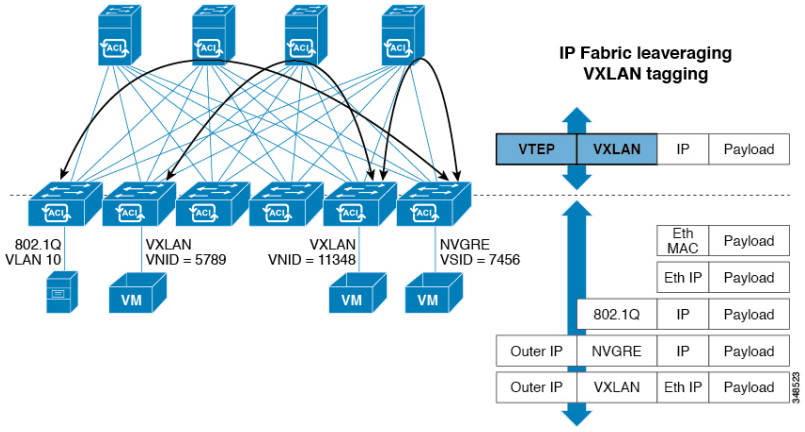
As traffic enters the fabric, ACI encapsulates and applies policy to it, forwards it as needed across the fabric through a spine switch (maximum two-hops), and de-encapsulates it upon exiting the fabric. Within the fabric, ACI uses Intermediate System-to-Intermediate System Protocol (IS-IS) and Council of Oracle Protocol (COOP) for all forwarding of endpoint to endpoint communications. This enables all ACI links to be active, equal cost multipath (ECMP) forwarding in the fabric, and fast-reconverging. For propagating routing information between software defined networks within the fabric and routers external to the fabric, ACI uses the Multiprotocol Border Gateway Protocol (MP-BGP).

VXLAN in ACI

VXLAN is an industry-standard protocol that extends Layer 2 segments over Layer 3 infrastructure to build Layer 2 overlay logical networks. The ACI infrastructure Layer 2 domains reside in the overlay, with isolated broadcast and failure bridge domains. This approach allows the data center network to grow without the risk of creating too large a failure domain.

All traffic in the ACI fabric is normalized as VXLAN packets. At ingress, ACI encapsulates external VLAN, VXLAN, and NVGRE packets in a VXLAN packet. The following figure shows ACI encapsulation normalization.

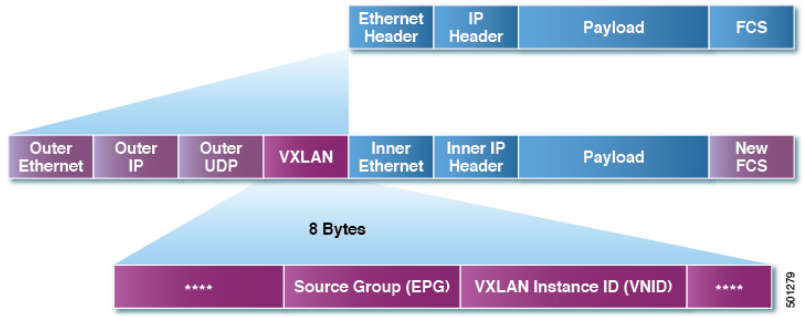
Figure 2: ACI Encapsulation Normalization



Forwarding in the ACI fabric is not limited to or constrained by the encapsulation type or encapsulation overlay network. An ACI bridge domain forwarding policy can be defined to provide standard VLAN behavior where required.

Because every packet in the fabric carries ACI policy attributes, ACI can consistently enforce policy in a fully distributed manner. ACI decouples application policy EPG identity from forwarding. The following illustration shows how the ACI VXLAN header identifies application policy within the fabric.

Figure 3: ACI VXLAN Packet Format



The ACI VXLAN packet contains both Layer 2 MAC address and Layer 3 IP address source and destination fields, which enables efficient and scalable forwarding within the fabric. The ACI VXLAN packet header source group field identifies the application policy endpoint group (EPG) to which the packet belongs. The VXLAN Instance ID (VNID) enables forwarding of the packet through tenant virtual routing and forwarding (VRF) domains within the fabric. The 24-bit VNID field in the VXLAN header provides an expanded address space for up to 16 million unique Layer 2 segments in the same network. This expanded address space gives IT departments and cloud providers greater flexibility as they build large multitenant data centers.

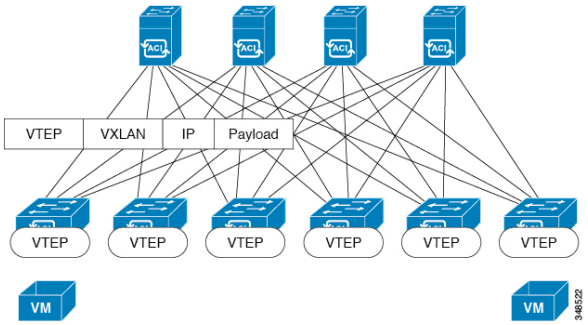
VXLAN enables ACI to deploy Layer 2 virtual networks at scale across the fabric underlay Layer 3 infrastructure. Application endpoint hosts can be flexibly placed in the data center network without concern for the Layer 3 boundary of the underlay infrastructure, while maintaining Layer 2 adjacency in a VXLAN overlay network.

Layer 3 VNIDs Facilitate Transporting Inter-subnet Tenant Traffic

The ACI fabric provides tenant default gateway functionality that routes between the ACI fabric VXLAN networks. For each tenant, the fabric provides a virtual default gateway that spans all of the leaf switches assigned to the tenant. It does this at the ingress interface of the first leaf switch connected to the endpoint. Each ingress interface supports the default gateway interface. All of the ingress interfaces across the fabric share the same router IP address and MAC address for a given tenant subnet.

The ACI fabric decouples the tenant endpoint address, its identifier, from the location of the endpoint that is defined by its locator or VXLAN tunnel endpoint (VTEP) address. Forwarding within the fabric is between VTEPs. The following figure shows decoupled identity and location in ACI.

Figure 4: ACI Decouples Identity and Location



VXLAN uses VTEP devices to map tenant end devices to VXLAN segments and to perform VXLAN encapsulation and de-encapsulation. Each VTEP function has two interfaces:

- A switch interface on the local LAN segment to support local endpoint communication through bridging
- An IP interface to the transport IP network

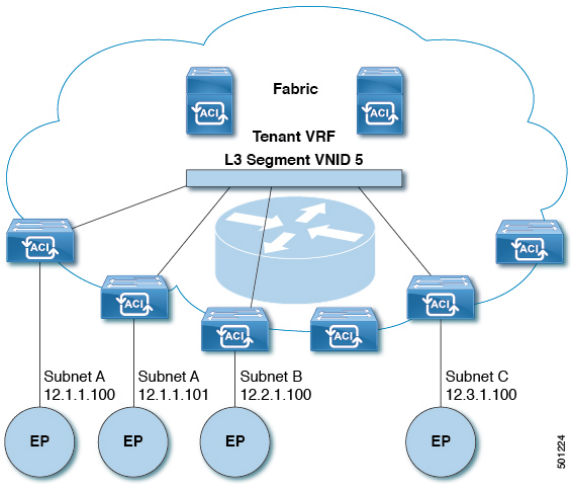
The IP interface has a unique IP address that identifies the VTEP device on the transport IP network known as the infrastructure VLAN. The VTEP device uses this IP address to encapsulate Ethernet frames and transmit the encapsulated packets to the transport network through the IP interface. A VTEP device also discovers the remote VTEPs for its VXLAN segments and learns remote MAC Address-to-VTEP mappings through its IP interface.

The VTEP in ACI maps the internal tenant MAC or IP address to a location using a distributed mapping database. After the VTEP completes a lookup, the VTEP sends the original data packet encapsulated in VXLAN with the destination address of the VTEP on the destination leaf switch. The destination leaf switch de-encapsulates the packet and sends it to the receiving host. With this model, ACI uses a full mesh, single hop, loop-free topology without the need to use the spanning-tree protocol to prevent loops.

The VXLAN segments are independent of the underlying network topology; conversely, the underlying IP network between VTEPs is independent of the VXLAN overlay. It routes the encapsulated packets 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.

The following figure shows how routing within the tenant is done.

Figure 5: Layer 3 VNIDs Transport ACI Inter-subnet Tenant Traffic



For each tenant VRF in the fabric, ACI assigns a single L3 VNID. ACI transports traffic across the fabric according to the L3 VNID. At the egress leaf switch, ACI routes the packet from the L3 VNID to the VNID of the egress subnet.

Traffic arriving at the fabric ingress that is sent to the ACI fabric default gateway is routed into the Layer 3 VNID. This provides very efficient forwarding in the fabric for traffic routed within the tenant. For example, with this model, traffic between 2 VMs belonging to the same tenant, on the same physical host, but on different subnets, only needs to travel to the ingress switch interface before being routed (using the minimal path cost) to the correct destination.

To distribute external routes within the fabric, ACI route reflectors use multiprotocol BGP (MP-BGP). The fabric administrator provides the autonomous system (AS) number and specifies the spine switches that become route reflectors.

Note Cisco ACI does not support IP fragmentation. Therefore, when you configure Layer 3 Outside (L3Out) connections to external routers, or Multi-Pod connections through an Inter-Pod Network (IPN), it is recommended that the interface MTU is set appropriately on both ends of a link.

IGP Protocol Packets (EIGRP, OSPFv3) are constructed by components based on the Interface MTU size. In Cisco ACI, if the CPU MTU size is less than the Interface MTU size and if the constructed packet size is greater than the CPU MTU, then the packet is dropped by the kernal, especially in IPv6. To avoid such control packet drops always configure the same MTU values on both the control plane and on the interface.

On some platforms, such as Cisco ACI, Cisco NX-OS, and Cisco IOS, the configurable MTU value does not take into account the Ethernet headers (matching IP MTU, and excluding the 14-18 Ethernet header size), while other platforms, such as IOS-XR, include the Ethernet header in the configured MTU value. A configured value of 9000 results in a max IP packet size of 9000 bytes in Cisco ACI, Cisco NX-OS, and Cisco IOS, but results in a max IP packet size of 8986 bytes for an IOS-XR untagged interface.

For the appropriate MTU values for each platform, see the relevant configuration guides.

We highly recommend that you test the MTU using CLI-based commands. For example, on the Cisco NX-OS CLI, use a command such as ping 1.1.1.1 df-bit packet-size 9000 source-interface ethernet 1/1.

Transmission of Spanning Tree Protocol BPDU

When there are two or more switches running the spanning tree protocol (STP) that are connected to Cisco Application Centric Infrastructure (ACI) in an EPG and if the static port assignments are as follows:

- All statically assigned ports under EPG are access untagged: STP bridge protocol data units (BPDUs) are received and sent without a tag.
- For a mix of statically assigned trunk ports and statically assigned access untagged ports: STP BPDUs received on a trunk port will be sent to the access untagged port with the `dot1q` tag. Hence the access port will go into a inconsistent state.
- For a mix of statically assigned trunk ports and statically assigned access ports under an EPG: Cisco ACI sends STP BPDU with `dot1q` tag and the access ports use `802.1p` access.

  In this case you must use `802.1p` access on the Layer 2 switch to receive and process the tagged STP packets.

  In case `802.1p` is not allowed on the Layer 2 switch, then use the trunk port access.
- Cisco ACI acts as a full-duplex hub and will flood spanning tree BPDUs within the VxLAN VNID associated to the encapsulation VLAN on which the BPDUs were received. Because Cisco ACI is a full-duplex medium, external switches running versions of rapid spanning tree protocol (RSTP) or rapid per VLAN spanning tree (RPVST) will default to point-to-point link types. As a result, if there are more than 2 external switches that run STP and connect to the same encapsulation VLAN and EPG VNID, you should set the link type to "shared" on the external switch interfaces to avoid problems with convergence and instability. These problems can occur because the switches will receive BPDUs from all bridges (or STP-enabled switches) that are connected to this encapsulation.

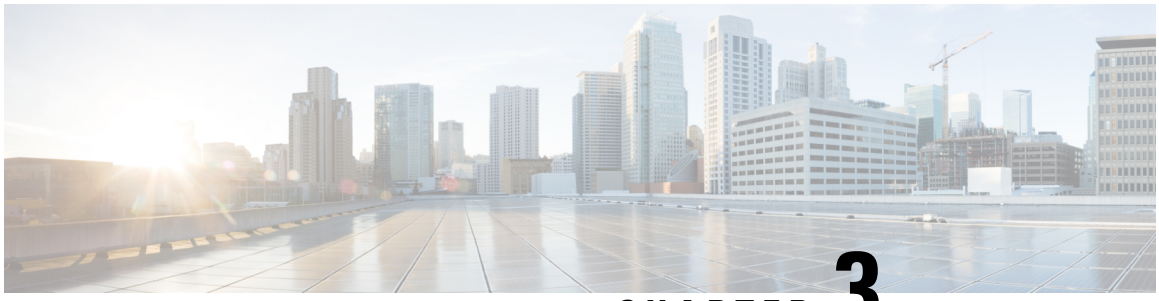
Spanning tree BPDUs are flooded within a given VLAN ID defined on an EPG path. This VLAN is referred to as `FD_VLAN` on leaf switches. To forward traffic within the FD_VLAN across leaf switches, Cisco ACI allocates a VXLAN VNID called the `fabric_encap`. The `fabric_encap` is derived by taking a numeric base identifier that belongs to a VLAN pool and then adding the index value for the VLAN ID that was allocated from the VLAN pool. For example, VXLAN VNID 9000 is assigned to the VLAN pool A that contains VLAN range 10-20. VLAN 10 from VLAN pool A is assigned VNID 9000, VLAN 11 is assigned VNID 9001.

For this reason, if two separate EPGs are using the same VLAN ID and are allocating that VLAN ID from the same VLAN pool, it is possible that the same `fabric_encap` VNID can be derived for two EPGs on different leaf switches. This creates the potential for spanning tree BPDUs to be unintentionally flooded between the two EPGs.

This behavior can be avoided by ensuring that different domains, such as physical domain, with separate VLAN pools are assigned to each EPG so that it allocates a given VLAN ID from separate VLAN pools. This ensures that the base identifier will be different and prevent overlapping `fabric_encap` VNIDs from being allocated.

The `fabric_encap` values can be checked using the following command and checking the "Fabric_enc" column in the output on leaf switches for a given 802.1q VLAN ID:

```
vsh_lc -c "show system internal eltmc info vlan br"
```



CHAPTER 3

Prerequisites for Configuring Layer 2 Networks

- Layer 2 Prerequisites, on page 9

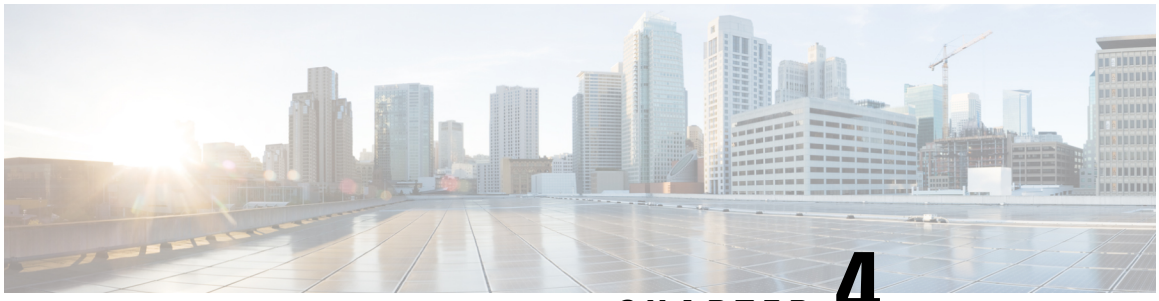
Layer 2 Prerequisites

Before you begin to perform the tasks in this guide, complete the following:

- Install the ACI fabric and ensure that the APIC controllers are online, and the APIC cluster is formed and healthy—For more information, see *Cisco APIC Getting Started Guide, Release 2.x*.
- Create fabric administrator accounts for the administrators that will configure Layer 2 networks—For instructions, see the *User Access, Authentication, and Accounting* and *Management* chapters in *Cisco APIC Basic Configuration Guide*.
- Install and register the target leaf switches in the ACI fabric—For more information, see *Cisco APIC Getting Started Guide, Release 2.x*.

  For information about installing and registering virtual switches, see *Cisco ACI Virtualization Guide*.
- Configure the tenants, VRFs, and EPGs (with application profiles and contracts) that will consume the Layer 2 networks—For instructions, see the *Basic User Tenant Configuration* chapter in *Cisco APIC Basic Configuration Guide*.

Caution If you install 1 Gigabit Ethernet (GE) or 10GE links between the leaf and spine switches in the fabric, there is risk of packets being dropped instead of forwarded, because of inadequate bandwidth. To avoid the risk, use 40GE or 100GE links between the leaf and spine switches.



CHAPTER 4

Networking Domains

This chapter contains the following sections:

- Networking Domains, on page 11
- Bridge Domains, on page 12
- VMM Domains, on page 12
- Configuring Physical Domains, on page 13

Networking Domains

A fabric administrator creates domain policies that configure ports, protocols, VLAN pools, and encapsulation. These policies can be used exclusively by a single tenant, or shared. Once a fabric administrator configures domains in the ACI fabric, tenant administrators can associate tenant endpoint groups (EPGs) to domains.

The following networking domain profiles can be configured:

- VMM domain profiles (`vmmDomP`) are required for virtual machine hypervisor integration.
- Physical domain profiles (`physDomP`) are typically used for bare metal server attachment and management access.
- Bridged outside network domain profiles (`l2extDomP`) are typically used to connect a bridged external network trunk switch to a leaf switch in the ACI fabric.
- Routed outside network domain profiles (`l3extDomP`) are used to connect a router to a leaf switch in the ACI fabric.
- Fibre Channel domain profiles (`fcDomP`) are used to connect Fibre Channel VLANs and VSANs.

A domain is configured to be associated with a VLAN pool. EPGs are then configured to use the VLANs associated with a domain.

Note EPG port and VLAN configurations must match those specified in the domain infrastructure configuration with which the EPG associates. If not, the APIC will raise a fault. When such a fault occurs, verify that the domain infrastructure configuration matches the EPG port and VLAN configurations.

Related Documents

For more information about Layer 3 Networking, see *Cisco APIC Layer 3 Networking Configuration Guide*.

For information about configuring VMM Domains, see *Cisco ACI Virtual Machine Networking* in *Cisco ACI Virtualization Guide.*

Bridge Domains

About Bridge Domains

A bridge domain (BD) represents a Layer 2 forwarding construct within the fabric. One or more endpoint groups (EPGs) can be associated with one bridge domain or subnet. A bridge domain can have one or more subnets that are associated with it. One or more bridge domains together form a tenant network. When you insert a service function between two EPGs, those EPGs must be in separate BDs. To use a service function between two EPGs, those EPGs must be isolated; this follows legacy service insertion based on Layer 2 and Layer 3 lookups.

VMM Domains

Virtual Machine Manager Domain Main Components

ACI fabric virtual machine manager (VMM) domains enable an administrator to configure connectivity policies for virtual machine controllers. The essential components of an ACI VMM domain policy include the following:

- **Virtual Machine Manager Domain Profile**—Groups VM controllers with similar networking policy requirements. For example, VM controllers can share VLAN pools and application endpoint groups (EPGs). The APIC communicates with the controller to publish network configurations such as port groups that are then applied to the virtual workloads. The VMM domain profile includes the following essential components:
  - **Credential**—Associates a valid VM controller user credential with an APIC VMM domain.
  - **Controller**—Specifes how to connect to a VM controller that is part of a policy enforcement domain. For example, the controller specifies the connection to a VMware vCenter that is part a VMM domain.



Note A single VMM domain can contain multiple instances of VM controllers, but they must be from the same vendor (for example, from VMware or from Microsoft.

- **EPG Association**—Endpoint groups regulate connectivity and visibility among the endpoints within the scope of the VMM domain policy. VMM domain EPGs behave as follows:
  - The APIC pushes these EPGs as port groups into the VM controller.

- An EPG can span multiple VMM domains, and a VMM domain can contain multiple EPGs.

- **Attachable Entity Profile Association**—Associates a VMM domain with the physical network infrastructure. An attachable entity profile (AEP) is a network interface template that enables deploying VM controller policies on a large set of leaf switch ports. An AEP specifies which switches and ports are available, and how they are configured.
- **VLAN Pool Association**—A VLAN pool specifies the VLAN IDs or ranges used for VLAN encapsulation that the VMM domain consumes.

Virtual Machine Manager Domains

An APIC VMM domain profile is a policy that defines a VMM domain. The VMM domain policy is created in APIC and pushed into the leaf switches.

VMM domains provide the following:

- A common layer in the ACI fabric that enables scalable fault-tolerant support for multiple VM controller platforms.
- VMM support for multiple tenants within the ACI fabric.

VMM domains contain VM controllers such as VMware vCenter or Microsoft SCVMM Manager and the credential(s) required for the ACI API to interact with the VM controller. A VMM domain enables VM mobility within the domain but not across domains. A single VMM domain can contain multiple instances of VM controllers but they must be the same kind. For example, a VMM domain can contain many VMware vCenters managing multiple controllers each running multiple VMs but it may not also contain SCVMM Managers. A VMM domain inventories controller elements (such as pNICs, vNICs, VM names, and so forth) and pushes policies into the controller(s), creating port groups, and other necessary elements. The ACI VMM domain listens for controller events such as VM mobility and responds accordingly.

Configuring Physical Domains

Configuring a Physical Domain

Physical domains control the scope of where a given VLAN namespace is used. The VLAN namespace that is associated with the physical domain is for non-virtualized servers, although it can also be used for static mapping of port-groups from virtualized servers. You can configure a physical domain for physical device types.

Before you begin

- Configure a tenant.

Procedure

Step 1 On the menu bar, click **Fabric**.

Step 2 On the submenu bar, click **External Access Policies**.

Step 3 In the **Navigation** pane, expand **Physical and External Domains** and click **Physical Domains**.

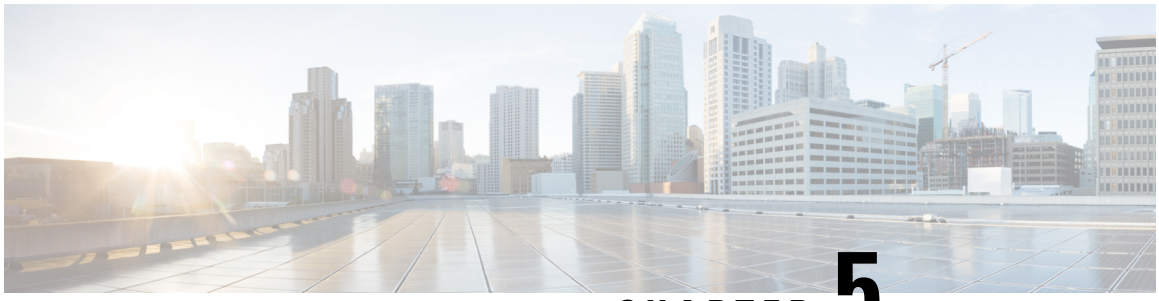
Step 4 From the **Actions** drop-down list, choose **Create Physical Domain**. The **Create Physical Domain** dialog box appears.

Step 5 Complete the following fields:

Name	Description
Name	The name of the physical domain profile.
Associate Attachable Entity Profiles	Choose the attachable entity profiles to be associated to this domain.
VLAN Pool	The VLAN pool used by the physical domain. The VLAN pool specifies the range or pool for VLANs that is allocated by the APIC for the service graph templates that are using this physical domain. Click **Dynamic** or **Static** allocation.

Step 6 (Optional) Add a AAA security domain and click the **Select** check box.

Step 7 Click **Submit**.



CHAPTER 5

Bridging

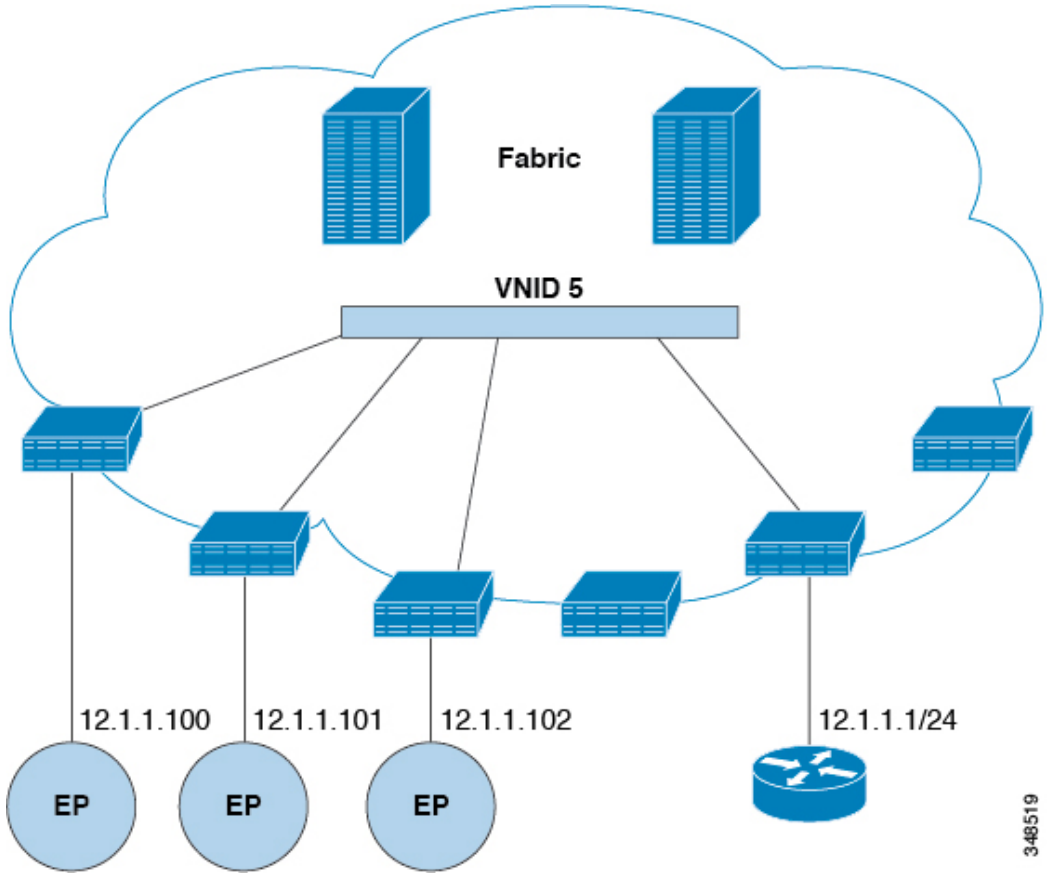
This chapter contains the following sections:

- Bridged Interface to an External Router, on page 15
- Bridge Domains and Subnets, on page 16
- Creating a Tenant, VRF, and Bridge Domain Using the GUI, on page 21
- Creating a Tenant, VRF, and Bridge Domain Using the NX-OS Style CLI, on page 23
- Configuring an Enforced Bridge Domain, on page 24
- Configuring Flood in Encapsulation for All Protocols and Proxy ARP Across Encapsulations, on page 26

Bridged Interface to an External Router

As shown in the figure below, when the leaf switch interface is configured as a bridged interface, the default gateway for the tenant VNID is the external router.

Figure 6: Bridged External Router

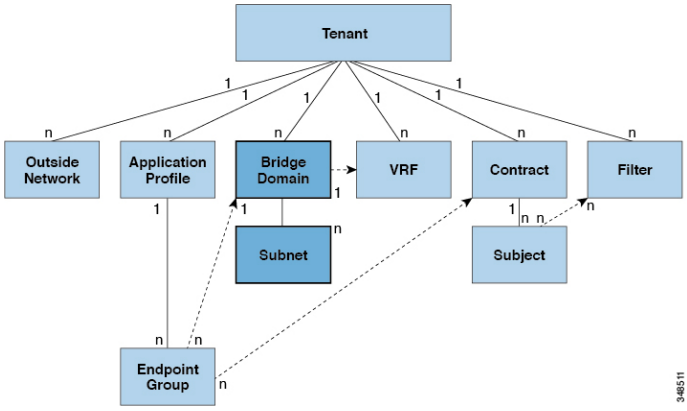


The ACI fabric is unaware of the presence of the external router and the APIC statically assigns the leaf switch interface to its EPG.

Bridge Domains and Subnets

A bridge domain (`fvBD`) represents a Layer 2 forwarding construct within the fabric. The following figure shows the location of bridge domains in the management information tree (MIT) and their relation to other objects in the tenant.

Figure 7: Bridge Domains



A bridge domain must be linked to a VRF instance (also known as a context or private network). With the exception of a Layer 2 VLAN, it must have at least one subnet (`fvSubnet`) associated with it. The bridge domain defines the unique Layer 2 MAC address space and a Layer 2 flood domain if such flooding is enabled. While a VRF instance defines a unique IP address space, that address space can consist of multiple subnets. Those subnets are defined in one or more bridge domains that reference the corresponding VRF instance.

The options for a subnet under a bridge domain or under an EPG are as follows:

- *Public*: The subnet can be exported to a routed connection.
- *Private*: The subnet applies only within its tenant.
- *Shared*: The subnet can be shared with and exported to multiple VRF instances in the same tenant or across tenants as part of a shared service. An example of a shared service is a routed connection to an EPG present in another VRF instance in a different tenant. This enables traffic to pass in both directions across VRF instances. An EPG that provides a shared service must have its subnet configured under that EPG (not under a bridge domain), and its scope must be set to advertised externally, and shared between VRF instances.

Note Shared subnets must be unique across the VRF instance involved in the communication. When a subnet under an EPG provides a Layer 3 external network shared service, such a subnet must be globally unique within the entire Cisco Application Centric Infrastructure (ACI) fabric.

Bridge domain packet behavior can be controlled in the following ways:

Packet Type	Mode
ARP	You can enable or disable **ARP Flooding**; without flooding, ARP packets are sent with unicast. **Note** If the `limitIpLearnToSubnets` in `fvBD` is set, endpoint learning is limited to the bridge domain only if the IP address is in a configured subnet of the bridge domain or an EPG subnet that is a shared service provider.
Unknown Unicast	**L2 Unknown Unicast**, which can be **Flood** or **Hardware Proxy**. **Note** When the bridge domain has **L2 Unknown Unicast** set to **Flood**, if an endpoint is deleted the system deletes it from both the local leaf switches as well as the remote leaf switches where the bridge domain is deployed, by selecting **Clear Remote MAC Entries**. Without this feature, the remote leaf continues to have this endpoint learned until the timer expires. Modifying the **L2 Unknown Unicast** setting causes traffic to bounce (go down and up) on interfaces to devices attached to EPGs associated with this bridge domain.
Unknown IP Multicast	**L3 Unknown Multicast Flooding** **Flood**: Packets are flooded on ingress and border leaf switch nodes only. With N9K-93180YC-EX, packets are flooded on all the nodes where a bridge domain is deployed. **Optimized**: Only 50 bridge domains per leaf are supported. This limitation is not applicable for N9K-93180YC-EX.
L2 Multicast, Broadcast, Unicast	**Multi-Destination Flooding**, which can be one of the following: • **Flood in BD**: Flood in bridge domain • **Flood in Encapsulation**: Flood in encapsulation • **Drop**: Drop the packets

Note Beginning with Cisco APIC release 3.1(1), on the Cisco Nexus 9000 series switches (with names ending with EX and FX and onwards), the following protocols can be flooded in encapsulation or flooded in a bridge domain: OSPF/OSPFv3, BGP, EIGRP, LACP, ISIS, IGMP, PIM, ST-BPDU, ARP/GARP, RARP, and ND.

Bridge domains can span multiple switches. A bridge domain can contain multiple subnets, but a subnet is contained within a single bridge domain. If the bridge domain (fvBD) `limitIPLearnToSubnets` property is set to `yes`, endpoint learning will occur in the bridge domain only if the IP address is within any of the configured subnets for the bridge domain or within an EPG subnet when the EPG is a shared service provider. Subnets can span multiple EPGs; one or more EPGs can be associated with one bridge domain or subnet. In hardware proxy mode, ARP traffic is forwarded to an endpoint in a different bridge domain when that endpoint has been learned as part of the Layer 3 lookup operation.

Bridge Domain Options

A bridge domain can be set to operate in flood mode for unknown unicast frames or in an optimized mode that eliminates flooding for these frames. When operating in flood mode, Layer 2 unknown unicast traffic is flooded over the multicast tree of the bridge domain (GIPo). For the bridge domain to operate in optimized mode you should set it to hardware-proxy. In this case, Layer 2 unknown unicast frames are sent to the spine-proxy anycast VTEP address.



Caution Changing from unknown unicast flooding mode to hw-proxy mode is disruptive to the traffic in the bridge domain.

If IP routing is enabled in the bridge domain, the mapping database learns the IP address of the endpoints in addition to the MAC address.

The **Layer 3 Configurations** tab of the bridge domain panel allows the administrator to configure the following parameters:

- **Unicast Routing**: If this setting is enabled and a subnet address is configured, the fabric provides the default gateway function and routes the traffic. Enabling unicast routing also instructs the mapping database to learn the endpoint IP-to-VTEP mapping for this bridge domain. The IP learning is not dependent upon having a subnet configured under the bridge domain.
- **Subnet Address**: This option configures the SVI IP addresses (default gateway) for the bridge domain.
- **Limit IP Learning to Subnet**: This option is similar to a unicast reverse-forwarding-path check. If this option is selected, the fabric will not learn IP addresses from a subnet other than the one configured on the bridge domain.



Caution Enabling **Limit IP Learning to Subnet** is disruptive to the traffic in the bridge domain.

Scaled L2 Only Mode - Legacy Mode

In Cisco Application Centric Infrastructure (ACI), the same VLAN ID can be reused for any purpose as long as the VLAN is deployed on different leaf nodes. This allows the Cisco ACI fabric to overcome the theoretical maximum number of VLANs 4094 as a fabric. However, to accomplish this, and also to hide the complexity of underlying VxLAN implementation, each individual leaf node can contain smaller number of VLANs. This may pose a problem when the density of VLANs per leaf node is required. In such a scenario, you can enable `Scaled L2 Only mode`, formerly known as legacy mode on the bridge domain. A bridge domain in

scaled L2 only mode allows large number of VLANs per leaf node. However, such a bridge domain has some limitations.

For the number of VLANs or bridge domains supported per leaf node with or without scaled L2 only mode, see Verified Scalability Guide for your specific release.

Limitations for Scaled L2 Only Mode

The following are limitations for legacy mode or scaled L2 only mode.

- The bridge domain can contain only one EPG and one VLAN.
- Unicast routing is not supported.
- Contracts are not supported.
- Dynamic VLAN allocation for VMM integration is not supported.
- Service graph is not supported.
- A QoS policy is not supported.
- The bridge domain essentially behaves as a VLAN in standalone Cisco NX-OS.

Scaled L2 Only Mode Configuration

The following are considerations to configure a bridge domain in scaled L2 only mode.

- VLAN ID is configured on the bridge domain.
- VLAN IDs configured under the EPG are overridden.
- Enabling or disabling a scaled L2 only mode on an existing bridge domain will impact service.

  Cisco Application Policy Infrastructure Controller (APIC) will automatically undeploy and redeploy the bridge domain when the VLAN ID is different from what was used prior to the change.

  When the same VLAN ID is used before and after the mode change, Cisco APIC will not automatically undeploy and redeploy the bridge domain. You must manually undeploy and redeply the bridge domain, which can be performed by deleting and recreating the static port configuration under the EPG.
- When changing the VLAN ID for scaled L2 only mode, you must first disable the mode, then enable scaled L2 only mode with the new VLAN ID.

Disabling IP Learning per Bridge Domain

You can disable IP dataplane learning for a bridge domain. The MAC learning still occurs in the hardware, but the IP learning only occurs from the ARP/GARP/ND processes. This functionality was introduced in the Cisco APIC 3.1 releases primarily for service graph policy-based redirect (PBR) deployments, and it has been superseded by the ability to disable IP dataplane learning per-VRF instance (Cisco APIC release 4.0), per bridge domain subnet (Cisco APIC release 5.2), and per-EPG (Cisco APIC release 5.2). We do not recommend disabling IP learning per bridge domain and it is not supported except when used with PBR.

See the following guidelines and limitations for disabling IP learning per bridge domain:

- Layer 3 multicast is not supported because the source IP address is not learned to populate the S,G information in the remote leaf switches.

- As the DL bit is set in the iVXLAN header, the MAC address is also not learned from the data path in the remote leaf switches. It results in flooding of the unknown unicast traffic from the remote leaf switch to all leaf switches in the fabric where this bridge domain is deployed. We recommend that you configure the bridge domain in proxy mode to overcome this situation if endpoint dataplane learning is disabled.
- ARP should be in flood mode and GARP based detection should be enabled.
- When IP learning is disabled, Layer 3 endpoints are not flushed in the corresponding VRF instance. It may lead to the endpoints pointing to the same leaf switch forever. To resolve this issue, flush all the remote IP endpoints in this VRF on all leaf switches.

The configuration change of disabling dataplane learning on the bridge domain does not flush previously locally learned endpoints. This limits the disruption to existing traffic flows. MAC learned endpoints age as usual if the Cisco ACI leaf switch sees no traffic with the given source MAC for longer than the endpoint retention policy.



Note Disabling IP dataplane learning means that the endpoint IP information is not updated as a result of traffic forwarding, but Cisco ACI can refresh the endpoint IP information with ARP/ND. This means that the aging of the local endpoints (whether they were learned before the configuration change, or they are learned after the configuration change) differs slightly from the normal aging and it depends also from `System > System Settings > Endpoint Controls > IP Aging`.

If `IP Aging` is disabled, traffic from a source MAC that matches an already learned endpoint MAC, refreshes the MAC addresses information in the endpoint table, and as a result also refreshes the IP information (this is the same as IP dataplane learning enabled).

If `IP Aging` is enabled, Cisco ACI ages out endpoint IP addresses individually (this is no different from what happens with IP dataplane learning enabled), but differently from configurations with IP dataplane learning enabled, traffic from a known source MAC and IP that matches an already learned endpoint, refreshes the MAC address information in the endpoint table, but not the IP information.

Creating a Tenant, VRF, and Bridge Domain Using the GUI

If you have a public subnet when you configure the routed outside, you must associate the bridge domain with the outside configuration.

Procedure

Step 1 On the menu bar, choose **Tenants** > **Add Tenant**.

Step 2 In the **Create Tenant** dialog box, perform the following tasks:

a) In the **Name** field, enter a name.
b) In the **Security Domains** section, click the **+** to open the **Create Security Domain** dialog box.
c) In the **Name** field, enter a name for the security domain, then click **Submit**.
d) In the **Create Tenant** dialog box, click **Update** for the security domain that you created.
e) Fill in the other fields as necessary.
f) Click **Submit**.

The ***tenant_name*** > **Networking** screen displays.

Step 3 In the **Work** pane, drag the **VRF** icon to the canvas to open the **Create VRF** dialog box, and perform the following tasks:

a) In the **Name** field, enter a name.
b) Fill in the other fields as necessary.
c) Click **Submit** to complete the VRF instance configuration.

Step 4 In the **Work** pane, drag the **Brdige Domain** icon to the canvas within the circle around the VRF instance to connect the two. In the **Create Bridge Domain** dialog box that displays, perform the following tasks:

a) In the **Name** field, enter a name.
b) Fill in the other fields as necessary.
c) Click **Next**.
d) In the **Subnets** section, click the **+** to open the **Create Subnet** dialog box.
e) In the **Gateway IP** field, enter the IP address and subnet mask.
f) Fill in the other fields as necessary.
g) Click **OK**.
h) Back in the **Create Bridge Domain** dialog box, fill in the other fields as necessary.
i) Click **Next**.
j) Fill in the fields as necessary.
k) Click **OK** to complete bridge domain configuration.

Step 5 In the **Work** pane, drag the **L3** icon to the canvas within the circle around the VRF instance to connect the two. In the **Create Routed Outside** dialog box that displays, perform the following tasks:

a) In the **Name** field, enter a name.
b) In the **Nodes And Interfaces Protocol Profiles** section, click the **+** to open the **Create Node Profile** dialog box.
c) In the **Name** field, enter a name.
d) In the **Nodes** section, click the **+** to open the **Select Node** dialog box.
e) In the **Node ID** drop-down list, choose a node.
f) In the **Router ID** field, enter the router ID.
g) In the **Static Routes** section, click the **+** to open the **Create Static Route** dialog box.
h) In the **Prefix** field, enter the IPv4 or IPv6 address.
i) In the **Next Hop Addresses** section, click the **+** to open the **Create Next Hop** dialog box.
j) In the **Next Hop Address** field, enter the IPv4 or IPv6 address.
k) In the **Preference** field, enter a number.
l) Fill in the other fields as necessary.
m) Click **OK**.
n) In the **Create Static Route** dialog box, fill in the other fields as necessary.
o) Click **OK**.
p) In the **Select Node** dialog box, fill in the other fields as necessary.
q) Click **OK**.
r) In the **Create Node Profile** dialog box, fill in the other fields as necessary.
s) Click **OK**.
t) Put a check in the **BGP**, **OSPF**, or **EIGRP** check boxes if desired.
u) Fill in the other fields as necessary.
v) Click **Next**.
w) Fill in the fields as necessary.
x) Click **OK** to complete the Layer 3 configuration.

To confirm the Layer 3 configuration, in the **Navigation** pane, expand **Networking** > **VRFs**.

Creating a Tenant, VRF, and Bridge Domain Using the NX-OS Style CLI

This section provides information on how to create tenants, VRFs, and bridge domains.



Note Before creating the tenant configuration, you must create a VLAN domain using the **vlan-domain** command and assign the ports to it.

Procedure

Step 1 Create a VLAN domain (which contains a set of VLANs that are allowable in a set of ports) and allocate VLAN inputs, as follows:

Example:

In the following example ("exampleCorp"), note that VLANs 50 - 500 are allocated.

```
apic1# configure
apic1(config)# vlan-domain dom_exampleCorp
apic1(config-vlan)# vlan 50-500
apic1(config-vlan)# exit
```

Step 2 Once the VLANs have been allocated, specify the leaf (switch) and interface for which these VLANs can be used. Then, enter "vlan-domain member" and then the name of the domain you just created.

Example:

In the following example, these VLANs (50 - 500) have been enabled on leaf 101 on interface ethernet 1/2-4 (three ports including 1/2, 1/3, and 1/4). This means that if you are using this interface, you can use VLANS 50-500 on this port for any application that the VLAN can be used for.

```
apic1(config-vlan)# leaf 101
apic1(config-vlan)# interface ethernet 1/2-4
apic1(config-leaf-if)# vlan-domain member dom_exampleCorp
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

Step 3 Create a tenant in global configuration mode, as shown in the following example:

Example:

```
apic1(config)# tenant exampleCorp
```

Step 4 Create a private network (also called VRF) in tenant configuration mode as shown in the following example:

Example:

```
apic1(config)# tenant exampleCorp
apic1(config-tenant)# vrf context exampleCorp_v1
apic1(config-tenant-vrf)# exit
```

Step 5 Create a bridge domain (BD) under the tenant, as shown in the following example:

Example:

```
apic1(config-tenant)# bridge-domain exampleCorp_b1
apic1(config-tenant-bd)# vrf member exampleCorp_v1
apic1(config-tenant-bd)# exit
```

Note
In this case, the VRF is "exampleCorp_v1".

Step 6 Allocate IP addresses for the BD (ip and ipv6), as shown in the following example.

Example:

```
apic1(config-tenant)# interface bridge-domain exampleCorp_b1
apic1(config-tenant-interface)# ip address 172.1.1.1/24
apic1(config-tenant-interface)# ipv6 address 2001:1:1::1/64
apic1(config-tenant-interface)# exit
```

What to do next

The next section describes how to add an application profile, create an application endpoint group (EPG), and associate the EPG to the bridge domain.

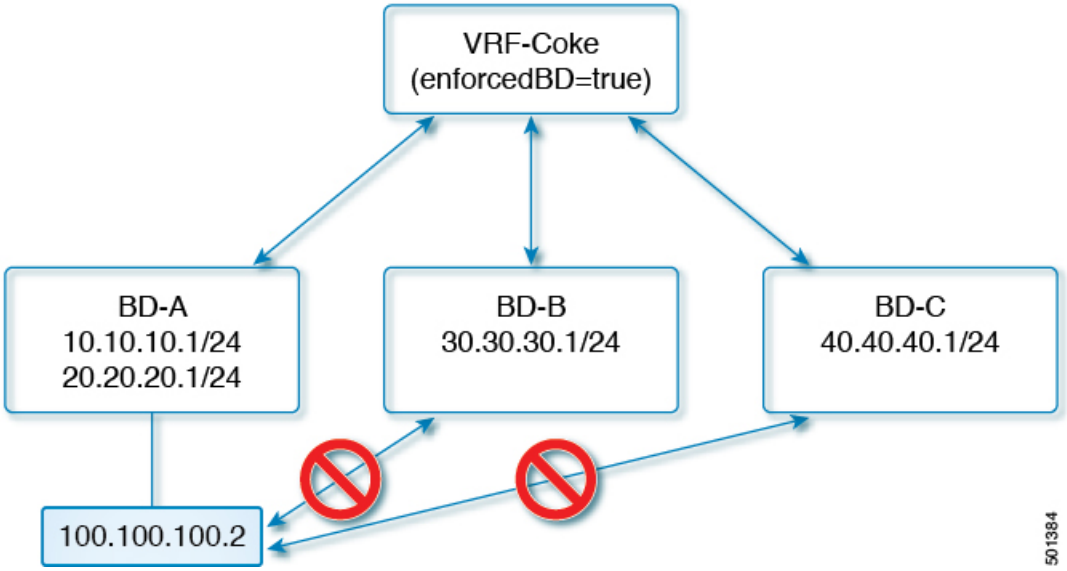
Related Topics

Configuring a VLAN Domain Using the NX-OS Style CLI

Configuring an Enforced Bridge Domain

An enforced bridge domain configuration entails creating an endpoint in a subject endpoint group (EPG) that can only ping subnet gateways within the associated bridge domain. With this configuration, you can then create a global exception list of IP addresses that can ping any subnet gateway.

Figure 8: Enforced Bridge Domain



Note

- The exception IP addresses can ping all of the bridge domain gateways across all of your VRF instances.
- A loopback interface configured for an L3Out does not enforce reachability to the IP address that is configured for the subject loopback interface.
- When an eBGP peer IP address exists in a different subnet than the subnet of the L3Out interface, you must add the peer subnet to the allowed exception subnets. Otherwise, eBGP traffic is blocked because the source IP address exists in a different subnet than the L3Out interface subnet.
- For a BGP prefixed-based peer, you must add the peer subnet to the list of allowed exception subnets. For example, if 20.1.1.0/24 is configured as BGP prefixed-based peer, you must add 20.1.1.0/24 to the list of allowed exception subnets.
- An enforced bridge domain is not supported with the Management tenant, regardless if the VRF instances are in-band or out-of-band, and any rules to control the traffic to these VRF instances should be configured using regular contracts.

Configuring an Enforced Bridge Domain Using the NX-OS Style CLI

This section provides information on how to configure your enforced bridge domain using the NX-OS style command line interface (CLI).

SUMMARY STEPS

1. Create and enable the tenant:
2. Add the subnet to the exception list.

DETAILED STEPS

Procedure

Step 1 Create and enable the tenant:

Example:

In the following example ("cokeVrf") is created and enabled.

```
apic1(config-tenant)# vrf context cokeVrf
apic1(config-tenant-vrf)# bd-enforce enable
apic1(config-tenant-vrf)# exit
apic1(config-tenant)#exit
```

Step 2 Add the subnet to the exception list.

Example:

```
apic1(config)#bd-enf-exp-ip add1.2.3.4/24
apic1(config)#exit
```

You can confirm if the enforced bridge domain is operational using the following type of command:

```
apic1# show running-config all | grep bd-enf
bd-enforce enable
bd-enf-exp-ip add 1.2.3.4/24
```

Example

The following command removes the subnet from the exception list:

```
apic1(config)# no bd-enf-exp-ip 1.2.3.4/24
apic1(config)#tenant coke
apic1(config-tenant)#vrf context cokeVrf
```

What to do next

To disable the enforced bridge domain run the following command:

```
apic1(config-tenant-vrf)# no bd-enforce enable
```

Configuring Flood in Encapsulation for All Protocols and Proxy ARP Across Encapsulations

Cisco Application Centric Infrastructure (ACI) uses the bridge domain as the Layer 2 broadcast boundary. Each bridge domain can include multiple endpoint groups (EPGs), and each EPG can be mapped to multiple

virtual or physical domains. Each EPG can also use different VLAN encapsulation pools in each domain. Each EPG can also use different VLAN or VXLAN encapsulation pools in each domain.

Ordinarily, when you put multiple EPGs within bridge domains, broadcast flooding sends traffic to all the EPGs in the bridge domain. Because EPGs are used to group endpoints and manage traffic to fulfill specific functions, sending the same traffic to all the EPGs in the bridge domain is not always practical.

The flood in encapsulation feature helps to consolidate bridge domains in your network. The feature does so by enabling you to control broadcast flooding to endpoints within the bridge domain based on the encapsulation of the virtual or physical domain that the EPGs are associated with.

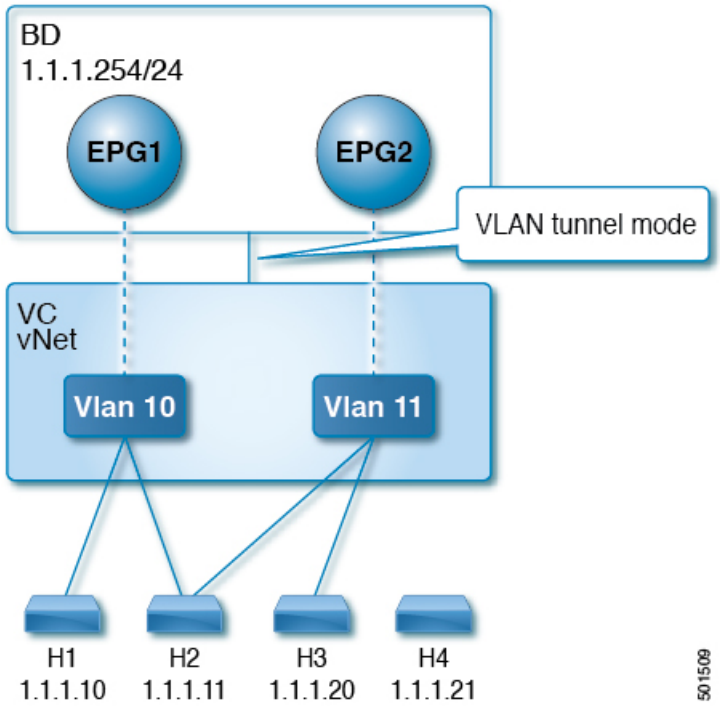
Flood in encapsulation requires the bridge domain to be configured with a subnet and with IP routing because in order to allow communication between endpoints of different EPGs in the same bridge domain Cisco ACI performs proxy ARP.

Example of Flood in Encapsulation Use Case with VLAN Encapsulation

Flood in encapsulation is often used when the external device is using Virtual Connect Tunnel mode where one MAC address is maintained per vNet because of VLAN-agnostic MAC learning.

Using multiple VLANs in tunnel mode can introduce a few challenges. In a typical deployment using Cisco ACI with a single tunnel, as illustrated in the following figure, there are multiple EPGs under one bridge domain. In this case, certain traffic is flooded within the bridge domain (and thus in all the EPGs), with the risk of MAC learning ambiguities that can cause forwarding errors.

Figure 9: Challenges of Cisco ACI with VLAN Tunnel Mode



In this topology, the blade switch (virtual connect in this example) has a single tunnel network defined that uses one uplink to connect with the Cisco ACI leaf node. Two user VLANs, VLAN 10 and VLAN 11 are carried over this link. The bridge domain is set in flooding mode as the servers' gateways are outside the Cisco ACI cloud. ARP negotiations occur in the following process:

- The server sends one ARP broadcast request over the VLAN 10 network.
- The ARP packet travels through the tunnel network to the external server, which records the source MAC address, learned from its downlink.
- The server then forwards the packet out its uplink to the Cisco ACI leaf switch.
- The Cisco ACI fabric sees the ARP broadcast packet entering on access port VLAN 10 and maps it to EPG1.
- Because the bridge domain is set to flood ARP packets, the packet is flooded within the bridge domain and thus to the ports under both EPGs as they are in the same bridge domain.
- The same ARP broadcast packet comes back over the same uplink.
- The blade switch sees the original source MAC address from this uplink.

Result: The blade switch has the same MAC address learned from both the downlink port and uplink port within its single MAC forwarding table, causing traffic disruptions.

Recommended Solution

The flood in encapsulation option is used to limit flooding traffic inside the bridge domain to a single encapsulation. When EPG1/VLAN X and EPG2/VLAN Y share the same bridge domain and flood in encapsulation is enabled, the encapsulation flooding traffic does not reach the other EPG/VLAN.

Beginning with Cisco Application Policy Infrastructure Controller (APIC) release 3.1(1), on the Cisco Nexus 9000 series switches (with names ending with EX and FX and onwards), all protocols are flooded in encapsulation. Also, when flood in encapsulation is enabled under the bridge domain for any inter-VLAN traffic, Proxy ARP ensures that the MAC flap issue does not occur. It also limits all flooding (ARP, GARP, and BUM) to the encapsulation. The restriction applies for all EPGs under the bridge domain where it is enabled.



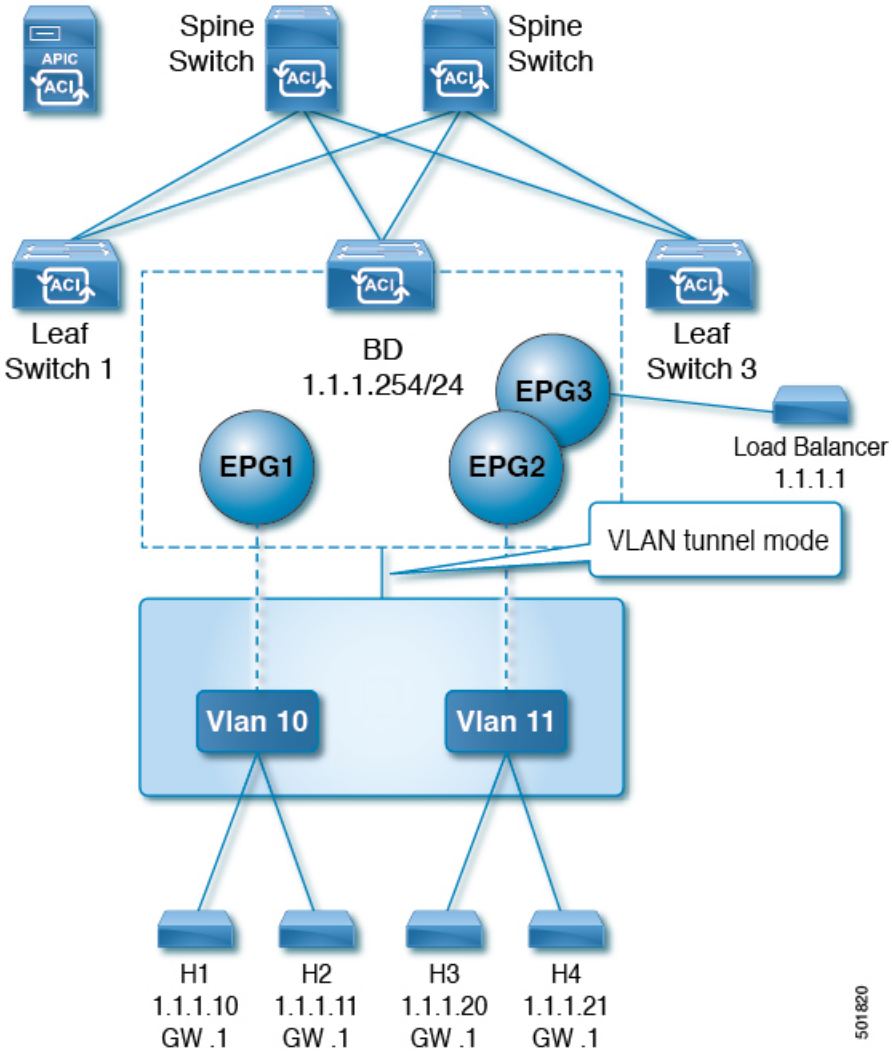
Note Before Cisco APIC release 3.1(1), these features are not supported (proxy ARP and all protocols being included when flooding within encapsulation). In an earlier Cisco APIC release or earlier generation switches (without EX or FX on their names), if you enable flood in encapsulation it does not function, no informational fault is generated, but Cisco APIC decreases the health score by 1.



Note Beginning with Cisco APIC release 3.2(5), you can configure flood in encapsulation for EPGs associated with VXLAN encapsulation. Previously, only VLANs were supported for flood in encapsulation for virtual domains. You configure flood in encapsulation when you create or modify a bridge domain or an EPG.

The recommended solution is to support multiple EPGs under one bridge domain by adding an external switch. This design with multiple EPGs under one bridge domain with an external switch is illustrated in the following figure.

Figure 10: Design with Multiple EPGs Under one Bridge Domain with an External Switch



Within the same bridge domain, some EPGs can be service nodes and other EPGs can have flood in encapsulation configured. A load balancer resides on a different EPG. The load balancer receives packets from the EPGs and sends them to the other EPGs (There is no Proxy ARP and flood within encapsulation does not take place).

Multi-Destination Protocol Traffic

The EPG/bridge domain level broadcast segmentation is supported for the following network control protocols:

- OSPF
- EIGRP
- LACP
- IS-IS
- BGP

- IGMP
- PIM
- STP-BPDU (flooded within EPG)
- ARP/GARP (controlled by ARP Proxy)
- ND

Flood in Encapsulation Limitations

The following limitations apply when using flood in encapsulation for all protocols:

- Flood in encapsulation does not work in ARP unicast mode.
- Neighbor Solicitation (Proxy NS/ND) is not supported for this release.
- Because proxy Address Resolution Protocol (ARP) is enabled implicitly, ARP traffic can go to the CPU for communication between different encapsulations.

  To ensure even distribution to different ports to process ARP traffic, enable per-port Control Plane Policing (CoPP) for ARP with flood in encapsulation.
- Flood in encapsulation is supported only in bridge domain in flood mode and ARP in flood mode. Bridge domain spine proxy mode is not supported.
- IPv4 Layer 3 multicast is not supported.
- IPv6 NS/ND proxy is not supported when flood in encapsulation is enabled. As a result, the connection between two endpoints that are under same IPv6 subnet but resident in EPGs with different encapsulation may not work.
- Virtual machine migration to a different VLAN has momentary issues (60 seconds). Virtual machine migration to a different VLAN or VXLAN has momentary issues (60 seconds).
- Setting up communication between virtual machines through a firewall, as a gateway, is not recommended because if the virtual machine IP address changes to the gateway IP address instead of the firewall IP address, then the firewall can be bypassed.
- Prior releases are not supported (even interoperating between prior and current releases).
- A mixed-mode topology with older-generation Application Leaf Engine (ALE) and Application Spine Engine (ASE) is not recommended and is not supported with flood in encapsulation. Enabling them together can prevent QoS priorities from being enforced.
- Flood in encapsulation is not supported for EPG and bridge domains that are extended across Cisco ACI fabrics that are part of the same Multi-Site domain. However, flood in encapsulation is still working and fully supported, and works for EPGs or bridge domains that are locally defined in Cisco ACI fabrics, independently from the fact those fabrics may be configured for Multi-Site. The same considerations apply for EPGs or bridge domains that are stretched between Cisco ACI fabric and remote leaf switches that are associated to that fabric.
- Flood in encapsulation is not supported on EPGs where any form of microsegmentation is configured. These include intra-EPG isolation, “allow microsegmentation” on a VMM domain, microsegmentation EPGs, and intra-EPG contracts.

- Flood in encapsulation is not supported for Common Pervasive Gateway. See the chapter "Common Pervasive Gateway" in the Cisco APIC Layer 3 Networking Configuration Guide.
- If you configure the flood in encapsulation on all EPGs of a bridge domain, ensure that you configure the flood in encapsulation on the bridge domain as well.
- IGMP snooping is not supported with flood in encapsulation.
- There is a condition that causes Cisco ACI to flood in the bridge domain (instead of the encapsulation) packets that are received on an EPG that is configured for flood in encapsulation. This happens regardless of whether the administrator configured flood in encapsulation directly on the EPG or on the bridge domain. The condition for this forwarding behavior is if the ingress leaf node has a remote endpoint for the destination MAC address while the egress leaf node does not have a corresponding local endpoint. This can happen due to reasons such as an interface flapping, an endpoint flush due to STP TCN, learning being disabled on the bridge domain due to an excessive amount of moves, and so on.

  In the 4.2(6o) and later 4.2(6) releases, 4.2(7m) and later 4.2(7) releases, and 5.2(1g) and later releases, this behavior was enhanced. If the administrator enables flood in encapsulation on the bridge domain (instead of the EPG), Cisco ACI does not send out such packets on any encapsulations from downlinks facing external devices on the non-ingress (egress and transit) leaf nodes. This new behavior prevents the packets from leaking to unexpected encapsulations. When flood in encapsulation is enabled only at an EPG level, the non-ingress leaf node may still flood packets in the bridge domain instead of the encapsulation. For more information, see the enhancement bug CSCvx83364.
- A Layer 3 gateway must be in the Cisco ACI fabric.

Configuring Flood in Encapsulation

You configure flood in encapsulation with the NX-OS style CLI, REST API, or the Cisco Application Policy Infrastructure Controller (APIC) GUI.

Flood in encapsulation that is configured for an EPG takes precedence over flood in encapsulation that is configured for a bridge domain (BD). When both BDs and EPGs are configured, the behavior is described as follows:

Table 4: Behavior When Both BDs and EPGs Are Configured

Configuration	Behavior
Flood in encapsulation at the EPG and flood in encapsulation at the bridge domain	Flood in encapsulation takes place for the traffic on all VLANs and VXLANs the bridge domain.
No flood in encapsulation at the EPG and flood in encapsulation at the bridge domain	Flood in encapsulation takes place for the traffic on all VLANs and VXLANs within the bridge domain.
Flood in encapsulation at the EPG and no flood in encapsulation at the bridge domain	Flood in encapsulation takes place for the traffic on that VLAN or VXLAN within the EPG of the bridge domain.
No flood in encapsulation at the EPG and no flood in encapsulation at the bridge domain	Flooding takes place within the entire bridge domain.

Configuring Flood in Encapsulation Using the Cisco APIC GUI

You configure flood in encapsulation using the Cisco Application Policy Infrastructure Controller (APIC) GUI when you create or modify a bridge domain (BD) or an endpoint group (EPG).

Procedure

Step 1 To configure flood in encapsulation while creating a BD, complete the following steps:

a) Log in to Cisco APIC.
b) Choose **Tenants** > ***tenant*** > **Networking** > **Bridge Domains**.
c) Right-click **Bridge Domains** and choose **Create Bridge Domain**.
d) In the **Create Bridge Domain** dialog box, Step 1, from the **Multi Destination Flooding** drop-down list, choose Flood in Encapsulation.
e) Fill out the other fields in the dialog box as appropriate to your setup, and click **Finish**.

Step 2 To configure flood in encapsulation while modifying a BD, complete the following steps:

a) Log in to Cisco APIC.
b) Go to **Tenants** > ***tenant*** > **Networking** > **Bridge Domains** > ***bridge domain***.
c) In the BD work pane, choose the **Policy** tab and then choose the **General** tab.
d) In the **Multi Destination Flooding** area, choose **Flood in Encapsulation**.
e) Click **Submit**.

Step 3 To configure flood in encapsulation while creating an EPG, complete the following steps:

a) Log in to Cisco APIC.
b) Go to **Tenants** > ***tenant*** > **Application Profiles**.
c) Right-click **Application Profiles** and then choose **Create Application EPG**.
d) In the **Create Application EPG** dialog box, in the **Flood in Encapsulation** area, choose **Enabled**.

Flood in encapsulation is disabled by default.

e) Fill out the other fields in the dialog box as appropriate to your setup, and click **Finish**.

Step 4 To configure flood in encapsulation while modifying an EPG, complete the following steps:

a) Go to **Tenants** > ***tenant*** > **Application Profiles** > **Application EPG** > ***application EPG***.
b) In the EPG work pane, choose the **Policy** tab and then choose the **General** tab.
c) In the **Flood in Encapsulation** area, choose **Enabled**.
d) Click **Submit**.

Configuring Flood in Encapsulation Using the NX-OS Style CLI

If you want to add flood in encapsulation only for selective endpoint groups (EPGs) using the NX-OS style CLI, enter the **flood-on-encapsulation enable** command under EPGs.

If you want to add flood in encapsulation for all EPGs, you can use the**multi-destination encap-flood** CLI command for the bridge domain.

Procedure

Step 1 Configure flood in encapsulation for the bridge domain (BD).

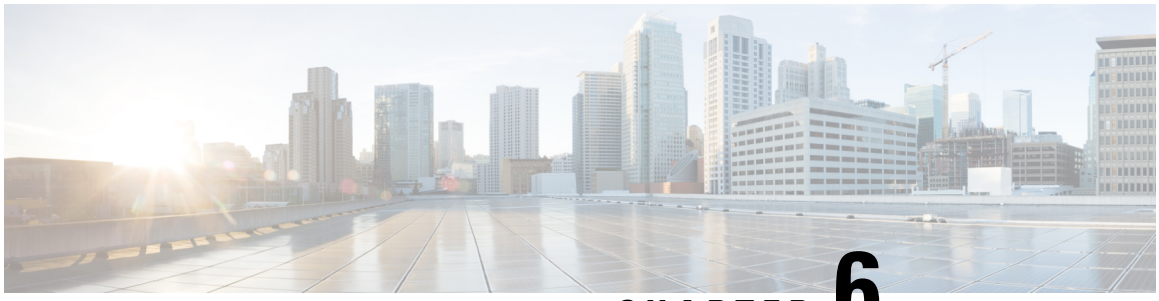
Example:

```
APIC1#configure
APIC1(config)# tenant tenant
APIC1(config-tenant)# bridge-domain BD-name
APIC1(config-tenant-bd)# multi-destination encap-flood
APIC1(config-tenant)#exit
APIC1(config)#
```

Step 2 Configure flood in encapsulation for the EPG.

Example:

```
APIC1(config)# tenant tenant
APIC1(config-tenant)# application AP1
APIC1(config-tenant-app)# epg EPG-name
APIC1(config-tenant-app-epg)# flood-on-encapsulation
APIC1(config-tenant-app-epg)#no flood-on-encapsulation
```



CHAPTER 6

EPGs

This chapter contains the following sections:

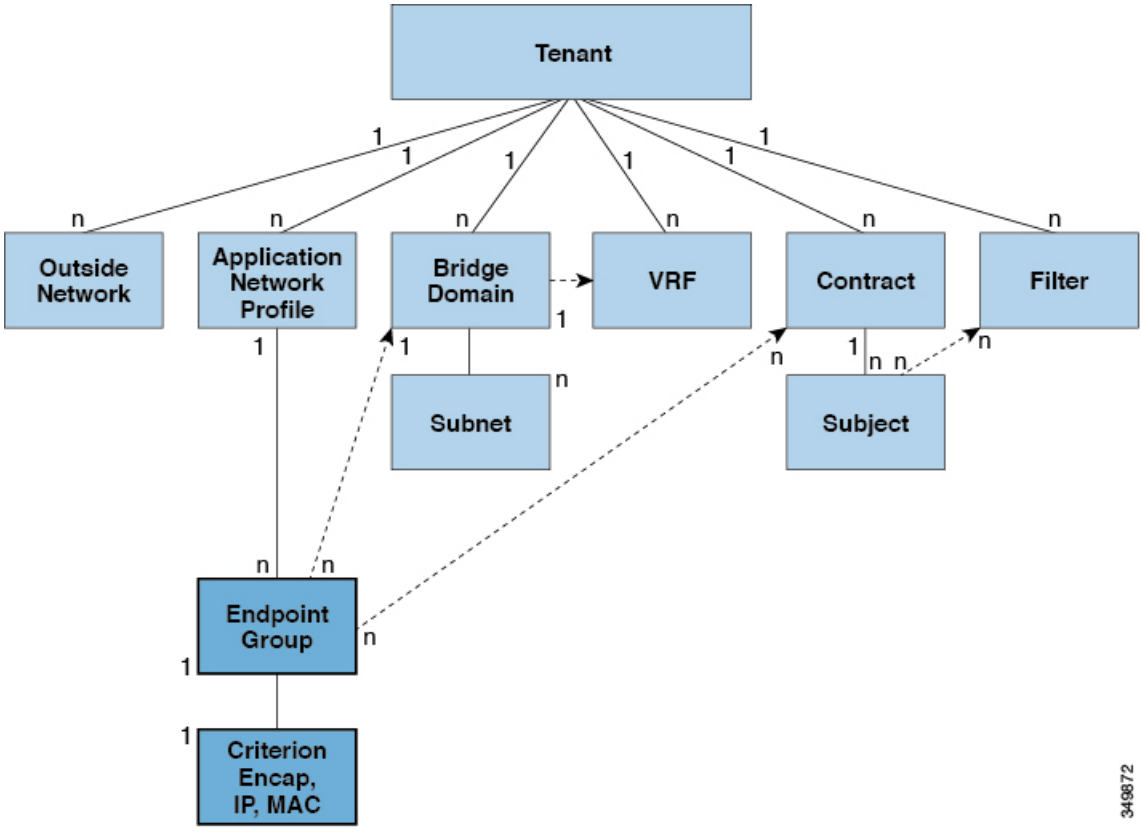
- About Endpoint Groups, on page 35
- Deploying an EPG on a Specific Port, on page 41
- Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port, on page 44
- Deploying EPGs to Multiple Interfaces Through Attached Entity Profiles, on page 47
- Intra-EPG Isolation, on page 50
- Configuring Intra-EPG Isolation for Cisco ACI Virtual Edge, on page 60
- Troubleshooting, on page 64
- Troubleshooting Endpoint Connectivity, on page 64
- Verifying IP-Based EPG Configurations, on page 68

About Endpoint Groups

Endpoint Groups

The endpoint group (EPG) is the most important object in the policy model. The following figure shows where application EPGs are located in the management information tree (MIT) and their relation to other objects in the tenant.

Figure 11: Endpoint Groups



An EPG is a managed object that is a named logical entity that contains a collection of endpoints. Endpoints are devices that are connected to the network directly or indirectly. They have an address (identity), a location, attributes (such as version or patch level), and can be physical or virtual. Knowing the address of an endpoint also enables access to all its other identity details. EPGs are fully decoupled from the physical and logical topology. Endpoint examples include servers, virtual machines, network-attached storage, or clients on the Internet. Endpoint membership in an EPG can be dynamic or static.

The ACI fabric can contain the following types of EPGs:

- Application endpoint group (`fvAEPg`)
- Layer 2 external outside network instance endpoint group (`l2extInstP`)
- Layer 3 external outside network instance endpoint group (`l3extInstP`)
- Management endpoint groups for out-of-band (`mgmtOoB`) or in-band (`mgmtInB`) access.

EPGs contain endpoints that have common policy requirements such as security, virtual machine mobility (VMM), QoS, or Layer 4 to Layer 7 services. Rather than configure and manage endpoints individually, they are placed in an EPG and are managed as a group.

Policies apply to EPGs, never to individual endpoints. An EPG can be statically configured by an administrator in the APIC, or dynamically configured by an automated system such as vCenter or OpenStack.

Note When an EPG uses a static binding path, the encapsulation VLAN associated with this EPG must be part of a static VLAN pool. For IPv4/IPv6 dual-stack configurations, the IP address property is contained in the `fvStIp` child property of the `fvStCEp` MO. Multiple `fvStIp` objects supporting IPv4 and IPv6 addresses can be added under one `fvStCEp` object. When upgrading ACI from IPv4-only firmware to versions of firmware that support IPv6, the existing IP property is copied to an `fvStIp` MO.

Regardless of how an EPG is configured, EPG policies are applied to the endpoints they contain.

WAN router connectivity to the fabric is an example of a configuration that uses a static EPG. To configure WAN router connectivity to the fabric, an administrator configures an `l3extInstP` EPG that includes any endpoints within an associated WAN subnet. The fabric learns of the EPG endpoints through a discovery process as the endpoints progress through their connectivity life cycle. Upon learning of the endpoint, the fabric applies the `l3extInstP` EPG policies accordingly. For example, when a WAN connected client initiates a TCP session with a server within an application (`fvAEPg`) EPG, the `l3extInstP` EPG applies its policies to that client endpoint before the communication with the `fvAEPg` EPG web server begins. When the client server TCP session ends and communication between the client and server terminate, that endpoint no longer exists in the fabric.

Note If a leaf switch is configured for *static binding (leaf switches)* under an EPG, the following restrictions apply:

- The static binding cannot be overridden with a static path.
- Interfaces in that switch cannot be used for routed external network (L3out) configurations.
- Interfaces in that switch cannot be assigned IP addresses.

Virtual machine management connectivity to VMware vCenter is an example of a configuration that uses a dynamic EPG. Once the virtual machine management domain is configured in the fabric, vCenter triggers the dynamic configuration of EPGs that enable virtual machine endpoints to start up, move, and shut down as needed.

ACI Policy Configuration in EPG Shutdown

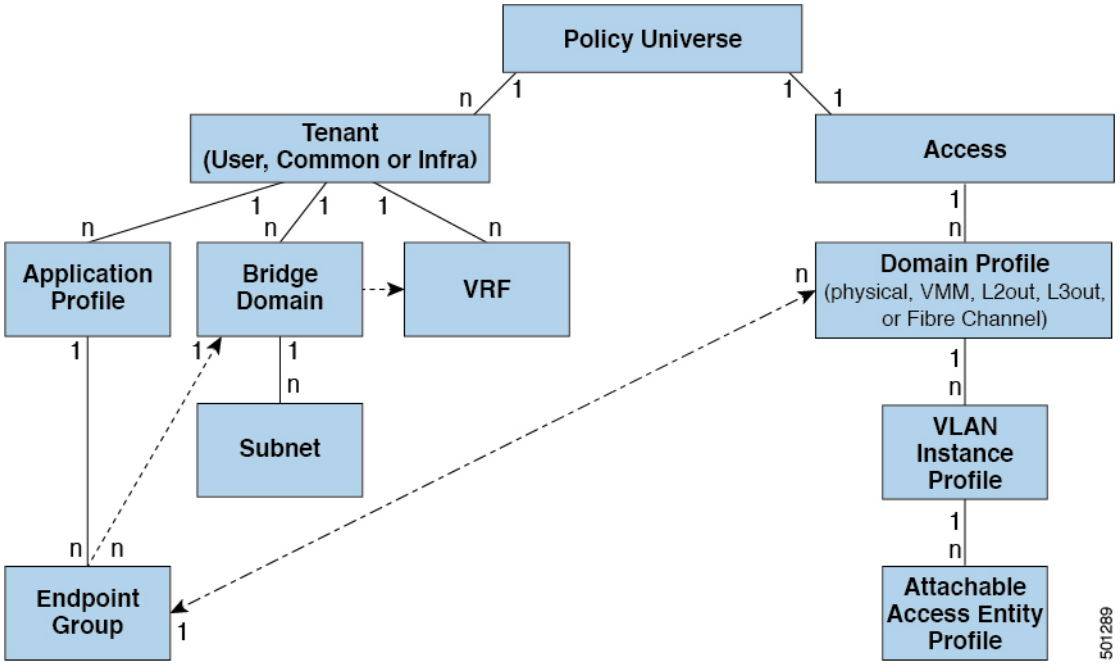
When the EPG is in shut down mode, the ACI policy configuration related to the EPG is removed from all the switches. The EPG is deleted from all the switches. While the EPG still exists in the ACI Data Store, it will be in inactive mode. In the APIC GUI you can check the box to remove the EPG from service.

Note Hosts attached to a EPG in shutdown mode cannot send or receive to/from the EPG.

Access Policies Automate Assigning VLANs to EPGs

While tenant network policies are configured separately from fabric access policies, tenant policies are not activated unless their underlying access policies are in place. Fabric access external-facing interfaces connect to external devices such as virtual machine controllers and hypervisors, hosts, routers, or Fabric Extenders (FEXs). Access policies enable an administrator to configure port channels and virtual port channels, protocols such as LLDP, CDP, or LACP, and features such as monitoring or diagnostics.

Figure 12: Association of Endpoint Groups with Access Policies



In the policy model, EPGs are tightly coupled with VLANs. For traffic to flow, an EPG must be deployed on a leaf port with a VLAN in a physical, VMM, L2out, L3out, or Fibre Channel domain. For more information, see Networking Domains, on page 11.

In the policy model, the domain profile associated to the EPG contains the VLAN instance profile. The domain profile contains both the VLAN instance profile (VLAN pool) and the attacheable Access Entity Profile (AEP), which are associated directly with application EPGs. The AEP deploys the associated application EPGs to all the ports to which it is attached, and automates the task of assigning VLANs. While a large data center could easily have thousands of active virtual machines provisioned on hundreds of VLANs, the ACI fabric can automatically assign VLAN IDs from VLAN pools. This saves a tremendous amount of time, compared with trunking down VLANs in a traditional data center.

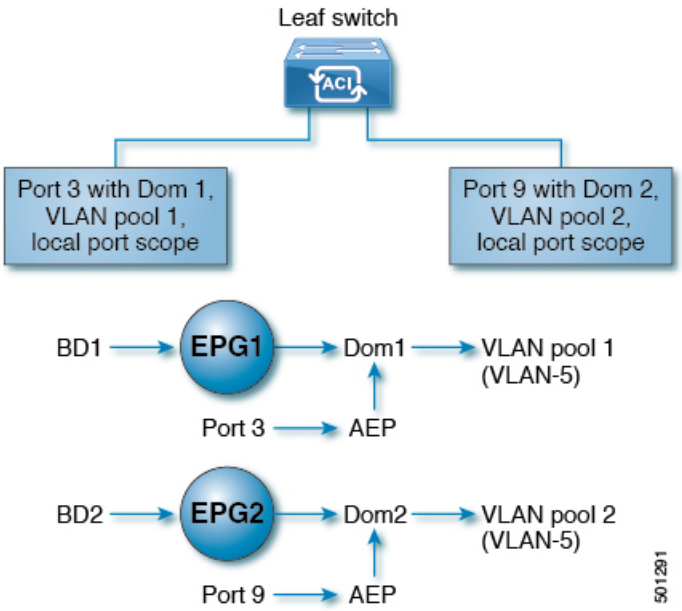
VLAN Guidelines

Use the following guidelines to configure the VLANs where EPG traffic will flow.

- Multiple domains can share a VLAN pool, but a single domain can only use one VLAN pool.
- To deploy multiple EPGs with same VLAN encapsulation on a single leaf switch, see Per Port VLAN, on page 38.

Per Port VLAN

You can deploy multiple EPGs with the same VLAN encapsulation on a given leaf switch (or FEX), in the Per Port VLAN configuration, similar to the following diagram:



Guidelines for per-port VLAN configuration

To enable deploying multiple EPGs using the same encapsulation number, on a single leaf switch, these guidelines apply:

- EPGs must be associated with different bridge domains.
- EPGs must be deployed on different ports.
- Both the port and EPG must be associated with the same domain that is associated with a VLAN pool that contains the VLAN number.
- Ports must be configured with `portLocal` VLAN scope.
- When an EPG has more than one physical domain with overlapping VLAN pools, avoid adding more than one domain to the AEP that is used to deploy the EPG on the ports. This avoids the risk of traffic forwarding issues.
- When an EPG has only one physical domain with overlapping VLAN pool, you can associate multiple domains with single AEP.

For example, with Per Port VLAN for the EPGs deployed on ports 3 and 9 in the diagram above, both using VLAN-5, port 3 and EPG1 are associated with Dom1 (pool 1) and port 9 and EPG2 are associated with Dom2 (pool 2).

Traffic coming from port 3 is associated with EPG1, and traffic coming from port 9 is associated with EPG2.

This does not apply to ports configured for Layer 3 external outside connectivity.

Limitations for per-port VLAN configuration

To enable deploying multiple EPGs using the same encapsulation number, on a single leaf switch, these limitations apply:

- Only ports that have the `vlanScope` set to `portlocal` allow allocation of separate (Port, VLAN) translation entries in both ingress and egress directions. For a given port with the `vlanScope` set to `portGlobal` (the default), each VLAN used by an EPG must be unique on a given leaf switch.
- Per Port VLAN is not supported on interfaces configured with Multiple Spanning Tree (MST), which requires VLAN IDs to be unique on a single leaf switch, and the VLAN scope to be global.
- Two interfaces cannot be configured with same VLAN pool if they are enabled with per-port VLAN confiugration.
- You cannot configure multiple interfaces with the same VLAN pool.
- You can have a maximum of 7,331,840 per port VLANs across the APIC fabric as the range is between 8,192 and 7,340,032. Therefore, we recommend you limit the VLAN pool size to the number of VLANs that you need for deploying EPGs on a single leaf switch.

Reusing VLAN Numbers Previously Used for EPGs on the Same Leaf Switch

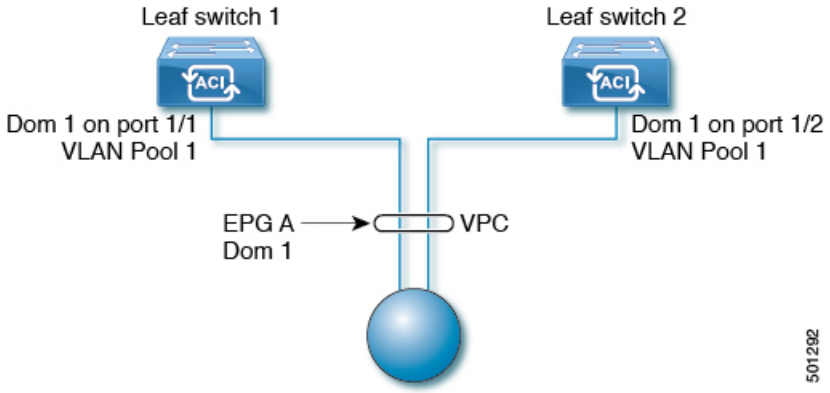
If you have previously configured VLANs for EPGs that are deployed on a leaf switch port, and you want to reuse the same VLAN numbers for different EPGs on different ports on the same leaf switch, use a process, such as the following example, to set them up without disruption:

In this example, EPGs were previously deployed on a port associated with a domain including a VLAN pool with a range of 9-100. You want to configure EPGs using VLAN encapsulations from 9-20.

1. Configure a new VLAN pool on a different port (with a range of, for example, 9-20).
2. Configure a new physical domain that includes leaf ports that are connected to firewalls.
3. Associate the physical domain to the VLAN pool you configured in step 1.
4. Configure the VLAN Scope as `portLocal` for the leaf port.
5. Associate the new EPGs (used by the firewall in this example) to the physical domain you created in step 2.
6. Deploy the EPGs on the leaf ports.

VLAN Guidelines for EPGs Deployed on vPCs

Figure 13: VLANs for Two Legs of a vPC



When an EPG is deployed on a vPC, it must be associated with the same domain (with the same VLAN pool) that is assigned to the leaf switch ports on the two legs of the vPC.

In this diagram, EPG A is deployed on a vPC that is deployed on ports on Leaf switch 1 and Leaf switch 2. The two leaf switch ports and the EPG are all associated with the same domain, containing the same VLAN pool.

Deploying an EPG on a Specific Port

Deploying an EPG on a Specific Node or Port Using the GUI

Before you begin

The tenant where you deploy the EPG is already created.

You can create an EPG on a specific node or a specific port on a node.

Procedure

Step 1 Log in to the Cisco APIC.

Step 2 Choose **Tenants** > ***tenant***.

Step 3 In the left navigation pane, expand ***tenant***, **Application Profiles**, and the ***application profile***.

Step 4 Right-click **Application EPGs** and choose **Create Application EPG**.

Step 5 In the **Create Application EPG STEP 1 > Identity** dialog box, complete the following steps:

a) In the **Name** field, enter a name for the EPG.

b) From the **Bridge Domain** drop-down list, choose a bridge domain.

c) Check the **Statically Link with Leaves/Paths** check box.

This check box allows you to specify on which port you want to deploy the EPG.

d) Click **Next**.

e) From the **Path** drop-down list, choose the static path to the destination EPG.

Step 6 In the **Create Application EPG STEP 2 > Leaves/Paths** dialog box, from the **Physical Domain** drop-down list, choose a physical domain.

Step 7 Complete one of the following sets of steps:

Option	Description
If you want to deploy the EPG on...	Then
A node	**a.** Expand the **Leaves** area. **b.** From the **Node** drop-down list, choose a node. **c.** In the **Encap** field, enter the appropriate VLAN. **d.** (Optional) From the **Deployment Immediacy** drop-down list, accept the default **On Demand** or choose **Immediate**.

Option	Description
	e. (Optional) From the Mode drop-down list, accept the default **Trunk** or choose another mode.
A port on the node	**a.** Expand the **Paths** area. **b.** From the **Path** drop-down list, choose the appropriate node and port. **c.** (Optional) In the **Deployment Immediacy** field drop-down list, accept the default **On Demand** or choose **Immediate**. **d.** (Optional) From the Mode drop-down list, accept the default **Trunk** or choose another mode. **e.** In the **Port Encap** field, enter the secondary VLAN to be deployed. **f.** (Optional) In the **Primary Encap** field, enter the primary VLAN to be deployed.

Step 8 Click **Update** and click **Finish**.

Step 9 In the left navigation pane, expand the EPG that you created.

Step 10 Complete one of the following actions:

- If you created the EPG on a node, click **Static Leafs**, and in the work pane view details of the static binding paths.
- If you created the EPG on a port of the node, click **Static Ports**, and in the work pane view details of the static binding paths.

Deploying an EPG on a Specific Port with APIC Using the NX-OS Style CLI

This procedure deploys an EPG on a specific port with Cisco Application Policy Infrastructure Controller (APIC) using the NX-OS-style CLI.

Note Do not mix using the GUI and the CLI when configuring per-interface on the Cisco APIC. Configurations that you perform in the GUI might only partially work in the NX-OS-style CLI.

Procedure

Step 1 Configure a VLAN domain:

Example:

```
apic1(config)# vlan-domain dom1
apic1(config-vlan)# vlan 10-100
```

Step 2 Create a tenant:

Example:

```
apic1# configure
apic1(config)# tenant t1
```

Step 3 Create a private network/VRF instance:

Example:

```
apic1(config-tenant)# vrf context ctx1
apic1(config-tenant-vrf)# exit
```

Step 4 Create a bridge domain:

Example:

```
apic1(config-tenant)# bridge-domain bd1
apic1(config-tenant-bd)# vrf member ctx1
apic1(config-tenant-bd)# exit
```

Step 5 Create an application profile and an application EPG:

Example:

```
apic1(config-tenant)# application AP1
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# bridge-domain member bd1
apic1(config-tenant-app-epg)# exit
apic1(config-tenant-app)# exit
apic1(config-tenant)# exit
```

Step 6 Associate the EPG with a specific port:

Example:

```
apic1(config)# leaf 1017
apic1(config-leaf)# interface ethernet 1/13
apic1(config-leaf-if)# vlan-domain member dom1
apic1(config-leaf-if)# switchport trunk allowed vlan 20 tenant t1 application AP1 epg EPG1
```

Note
The vlan-domain and vlan-domain member commands in the example are a prerequisite for deploying an EPG on a port.

Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port

Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port

This topic provides a typical example of how to create physical domains, Attach Entity Profiles (AEP), and VLANs that are mandatory to deploy an EPG on a specific port.

All endpoint groups (EPGs) require a domain. Interface policy groups must also be associated with Attach Entity Profile (AEP), and the AEP must be associated with a domain, if the AEP and EPG have to be in same domain. Based on the association of EPGs to domains and of interface policy groups to domains, the ports and VLANs that the EPG uses are validated. The following domain types associate with EPGs:

- Application EPGs
- Layer 3 external outside network instance EPGs
- Layer 2 external outside network instance EPGs
- Management EPGs for out-of-band and in-band access

The APIC checks if an EPG is associated with one or more of these types of domains. If the EPG is not associated, the system accepts the configuration but raises a fault. The deployed configuration may not function properly if the domain association is not valid. For example, if the VLAN encapsulation is not valid for use with the EPG, the deployed configuration may not function properly.



Note EPG association with the AEP without static binding does not work in a scenario when you configure the EPG as **Trunk** under the AEP with one end point under the same EPG supporting Tagging and the other end point in the same EPG does not support VLAN tagging. While associating AEP under the EPG, you can configure it as Trunk, Access (Tagged) or Access (Untagged).

Creating Domains, and VLANs to Deploy an EPG on a Specific Port Using the GUI

Before you begin

- The tenant where you deploy the EPG is already created.
- An EPG is statically deployed on a specific port.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the **Navigation** pane, choose **Quick Start**.

Step 3 In the **Work** pane, click **Configure Interfaces**.

Step 4 In the **Configure Interfaces** dialog, perform the following actions:

a) For **Node Type**, click **Leaf**.
b) For **Port Type**, click **Access**.
c) For **Interface Type**, choose the desired type.
d) For **Interface Aggregation Type**, choose **Individual**.
e) For **Node**, click **Select Node**, put a check in the box for the desired node, then click **OK**. You can select multiple nodes.
f) For **Interfaces For All Switches**, enter the range of desired interfaces.
g) For **Leaf Access Port Policy Group**, click **Select Leaf Access Port Policy Group**.
h) In the **Select Leaf Access Port Policy Group** dialog, click **Create Leaf Access Port Policy Group**.
i) In the **Create Leaf Access Port Policy Group** dialog, for **Link Level Policy**, click **Select Link Level Policy**.
j) Choose a link level policy and click **Select**, or click **Create Link Level Policy**, fill out the fields as desired, and click **Save**.
k) Click **Save**.

Step 5 Create a domain and VLAN pool by performing the following actions:

a) In the **Navigation** pane, expand **Physical and External Domains**.
b) Right-click **Physical Domains** and choose the appropriate **Create Physical Domain**.
c) For **Name**, enter a name for the domain.
d) For **VLAN Pool**, choose **Create VLAN Pool**, fill out the fields as desired, then click **Submit**.
e) Fill out the remaining fields as desired.
f) Click **Submit**.

Step 6 On the menu bar, choose **Tenants** > **All Tenants**.

Step 7 In the **Work** pane, double click the desired tenant.

Step 8 In the **Navigation** pane, expand ***Tenant_name*** > **Application Profiles** > ***profile_name*** > **Application EPGs** > ***EPG_name*** and perform the following actions:

a) Right-click **Domains (VMs and Bare-Metals)** and choose **Add Physical Domain Association**.
b) In the **Add Physical Domain Association** dialog, from the **Physical Domain Profile** drop-down list, choose the domain that you created.
c) Click **Submit**.
The AEP is associated with a specific port on a node and with a domain. The physical domain is associated with the VLAN pool and the tenant is associated with this physical domain.

The switch profile and the interface profile are created. The policy group is created in the port block under the interface profile. The AEP is automatically created, and it is associated with the port block and with the domain. The domain is associated with the VLAN pool and the tenant is associated with the domain.

Creating AEP, Domains, and VLANs to Deploy an EPG on a Specific Port Using the NX-OS Style CLI

Before you begin

- The tenant where you deploy the EPG is already created.
- An EPG is statically deployed on a specific port.

Procedure

Step 1 Create a VLAN domain and assign VLAN ranges:

Example:

```
apic1(config)# vlan-domain domP
apic1(config-vlan)# vlan 10
apic1(config-vlan)# vlan 25
apic1(config-vlan)# vlan 50-60
apic1(config-vlan)# exit
```

Step 2 Create an interface policy group and assign a VLAN domain to the policy group:

Example:

```
apic1(config)# template policy-group PortGroup
apic1(config-pol-grp-if)# vlan-domain member domP
```

Step 3 Create a leaf interface profile, assign an interface policy group to the profile, and assign the interface IDs on which the profile will be applied:

Example:

```
apic1(config)# leaf-interface-profile InterfaceProfile1
apic1(config-leaf-if-profile)# leaf-interface-group range
apic1(config-leaf-if-group)# policy-group PortGroup
apic1(config-leaf-if-group)# interface ethernet 1/11-13
apic1(config-leaf-if-profile)# exit
```

Step 4 Create a leaf profile, assign the leaf interface profile to the leaf profile, and assign the leaf IDs on which the profile will be applied:

Example:

```
apic1(config)# leaf-profile SwitchProfile-1019
apic1(config-leaf-profile)# leaf-interface-profile InterfaceProfile1
apic1(config-leaf-profile)# leaf-group range
apic1(config-leaf-group)# leaf 1019
apic1(config-leaf-group)#
```

Validating Overlapping VLANs

This global feature prevents association of overlapping VLAN pools on a single EPG. If there are any overlapping pools allocated with any EPG in APIC, then this feature cannot be enabled (an error is displayed if there is an attempt to enable it). If no existing overlapping pools are present, then this feature can be enabled. Once enabled, when an attempt to allocate a domain on an EPG is performed, and the domain contains a VLAN pool with a range overlapping with another domain already associate to the EPG, then the configuration is blocked.

When overlapping VLAN pools exist under an EPG, then the FD VNID allocated for the EPG by each switch is non-deterministic and different switches may allocate different VNIDs. This can cause EPM sync failures between leafs within a vPC domain (causing intermittent connectivity for all endpoints within the EPG). It can also cause bridging loops if user is extending STP between the EPG, as the BPDUs will be dropped between switches due to FD VNID mismatch.

Validating Overlapping VLANs Using the GUI

This procedure provides an example of using the APIC GUI to configure overlapping VLAN validation.

Procedure

Step 1 On the menu bar, choose **System** > **System Settings**.

Step 2 In the navigation pane, choose **Fabric Wide Setting**.

Step 3 In the work pane, locate and check **Enforce EPG VLAN Validation**.

Note
If overlapping VLAN pools already exist and this parameter is checked, the system returns an error. You must assign VLAN pools that are not overlapping to the EPGs before choosing this feature.

If this parameter is checked and an attempt is made to add an overlapping VLAN pool to an EPG, the system returns an error.

Step 4 Click **Submit**.

Deploying EPGs to Multiple Interfaces Through Attached Entity Profiles

Deploying an Application EPG through an AEP or Interface Policy Group to Multiple Ports

Through the APIC Advanced GUI and REST API, you can associate attached entity profiles directly with application EPGs. By doing so, you deploy the associated application EPGs to all those ports associated with the attached entity profile in a single configuration.

Through the APIC REST API or the NX-OS style CLI, you can deploy an application EPG to multiple ports through an Interface Policy Group.

Deploying an EPG through an AEP to Multiple Interfaces Using the APIC GUI

You can quickly associate an application with an attached entity profile to quickly deploy that EPG over all the ports associated with that attached entity profile.

Before you begin

- The target application EPG is created.
- The VLAN pools has been created containing the range of VLANs you wish to use for EPG Deployment on the AEP.
- The physical domain has been created and linked to the VLAN Pool and AEP.
- The target attached entity profile is created and is associated with the ports on which you want to deploy the application EPG.

Procedure

Step 1 Navigate to the target attached entity profile.

a) Open the page for the attached entity profile to use. In the GUI, click **Fabric** > **Access Policies** > **Policies** > **Global** > **Attachable Access Entity Profiles**.

b) Click the target attached entity profile to open its Attachable Access Entity Profile window.

Step 2 Click the **Show Usage** button to view the leaf switches and interfaces associated with this attached entity profile.

the application EPGs associated with this attached entity profile are deployed to all the ports on all the switches associated with this attached entity profile.

Step 3 Use the **Application EPGs** table to associate the target application EPG with this attached entity profile. Click + to add an application EPG entry. Each entry contains the following fields:

Field	Action
Application EPGs	Use the drop down to choose the associated Tenant, Application Profile, and target application EPG.
Encap	Enter the name of the VLAN over which the target application EPG will communicate.
Primary Encap	If the application EPG requires a primary VLAN, enter the name of the primary VLAN.
Mode	Use the drop down to specify the mode in which data is transmitted: • **Trunk** -- Choose if traffic from the host is tagged with a VLAN ID. • **Access** -- Choose if traffic from the host is tagged with an 802.1p tag. • **Access Untagged** -- Choose if the traffic from the host is untagged.

Step 4 Click **Submit**.
the application EPGs associated with this attached entity profile are deployed to all the ports on all the switches associated with this attached entity profile.

Deploying an EPG through an Interface Policy Group to Multiple Interfaces Using the NX-OS Style CLI

In the NX-OS CLI, an attached entity profile is not explicitly defined to associate with an EPG for rapid deployment; instead the interface policy group is defined, assigned a domain, applied to all the ports associated with a VLAN and configured to include the application EPG to be deployed over that VLAN.

Before you begin

- The target application EPG is created.
- The VLAN pools has been created containing the range of VLANs you wish to use for EPG Deployment on the AEP.
- The physical domain has been created and linked to the VLAN Pool and AEP.
- The target attached entity profile is created and is associated with the ports on which you want to deploy the application EPG.

Procedure

Step 1 Associate the target EPG with the interface policy group.

The sample command sequence specifies an interface policy group **pg3** associated with VLAN domain, **domain1**, and with VLAN **1261**. The application EPG, **epg47** is deployed to all interfaces associated with this policy group.

Example:

```
 apic1# configure terminal
apic1(config)# template policy-group pg3
apic1(config-pol-grp-if)# vlan-domain member domain1
apic1(config-pol-grp-if)# switchport trunk allowed vlan 1261 tenant tn10 application pod1-AP
      epg epg47
```

Step 2 Check the target ports to ensure deployment of the policies of the interface policy group associated with application EPG.

The output of the sample **show** command sequence indicates that policy group **pg3** is deployed on Ethernet port **1/20** on leaf switch **1017**.

Example:

```
 apic1# show run leaf 1017 int eth 1/20
# Command: show running-config leaf 1017 int eth 1/20
# Time: Mon Jun 27 22:12:10 2016
  leaf 1017
    interface ethernet 1/20
      policy-group pg3
      exit
    exit
```

```
ifav28-ifc1#
```

Intra-EPG Isolation

Intra-EPG Endpoint Isolation

Intra-EPG endpoint isolation policies provide full isolation for virtual or physical endpoints; no communication is allowed between endpoints in an EPG that is operating with isolation enforced. Isolation enforced EPGs reduce the number of EPG encapsulations required when many clients access a common service but are not allowed to communicate with each other.

An EPG is isolation enforced for all Cisco Application Centric Infrastructure (ACI) network domains or none. While the Cisco ACI fabric implements isolation directly to connected endpoints, switches connected to the fabric are made aware of isolation rules according to a primary VLAN (PVLAN) tag.



Note If an EPG is configured with intra-EPG endpoint isolation enforced, these restrictions apply:

- All Layer 2 endpoint communication across an isolation enforced EPG is dropped within a bridge domain.
- All Layer 3 endpoint communication across an isolation enforced EPG is dropped within the same subnet.
- Preserving QoS CoS priority settings is not supported when traffic is flowing from an EPG with isolation enforced to an EPG without isolation enforced.

BPDUs are not forwarded through EPGs with intra-EPG isolation enabled. Therefore, when you connect an external Layer 2 network that runs spanning tree in a VLAN that maps to an isolated EPG on Cisco ACI, Cisco ACI might prevent spanning tree in the external network from detecting a Layer 2 loop. You can avoid this issue by ensuring that there is only a single logical link between Cisco ACI and the external network in these VLANs.

Intra-EPG Isolation for Bare Metal Servers

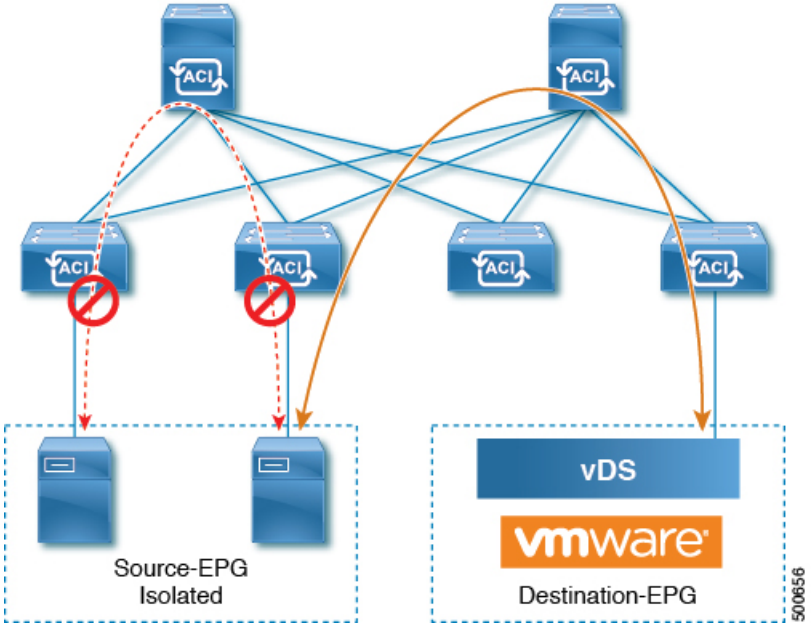
Intra-EPG Isolation for Bare Metal Servers

Intra-EPG endpoint isolation policies can be applied to directly connected endpoints such as bare metal servers.

Examples use cases include the following:

- Backup clients have the same communication requirements for accessing the backup service, buy they don't need to communicate with each other.
- Servers behind a load balancer have the same communication requirements, but isolating them from each other protects against a server that is compromised or infected.

Figure 14: Intra-EPG Isolation for Bare Metal Servers



Bare metal EPG isolation is enforced at the leaf switch. Bare metal servers use VLAN encapsulation. All unicast, multicast and broadcast traffic is dropped (denied) within isolation enforced EPGs. ACI bridge-domains can have a mix of isolated and regular EPGs. Each Isolated EPG can have multiple VLANs where intra-vlan traffic is denied.

Configuring Intra-EPG Isolation for Bare Metal Servers Using the GUI

The port the EPG uses must be associated with a bare metal server interface in the physical domain that is used to connect the bare metal servers directly to leaf switches.

SUMMARY STEPS

1. In a tenant, right click on an **Application Profile**, and open the **Create Application EPG** dialog box to perform the following actions:
2. In the **Leaves/Paths** dialog box, perform the following actions:

DETAILED STEPS

Procedure

Step 1 In a tenant, right click on an **Application Profile**, and open the **Create Application EPG** dialog box to perform the following actions:

a) In the **Name** field, add the EPG name (intra_EPG-deny).
b) For **Intra EPG Isolation**, click **Enforced**.
c) In the **Bridge Domain** field, choose the bridge domain from the drop-down list (bd1).
d) Check the **Statically Link with Leaves/Paths** check box.
e) Click **Next**.

Step 2 In the **Leaves/Paths** dialog box, perform the following actions:

a) In the **Path** section, choose a path from the drop-down list (Node-107/eth1/16) in Trunk Mode.

Specify the **Port Encap** (vlan-102) for the secondary VLAN.

Note
If the bare metal server is directly connected to a leaf switch, only the Port Encap secondary VLAN is specified.

Specify the **Primary Encap** (vlan-103) for the primary VLAN.

b) Click **Update**.

c) Click **Finish**.

Configuring Intra-EPG Isolation for Bare Metal Servers Using the NX-OS Style CLI

SUMMARY STEPS

1. In the CLI, create an intra-EPG isolation EPG:
2. Verify the configuration:

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	In the CLI, create an intra-EPG isolation EPG:
	Example:
	The VMM case is below.
	`ifav19-ifc1(config)# tenant Test_Isolation`
	`ifav19-ifc1(config-tenant)# application PVLAN`
	`ifav19-ifc1(config-tenant-app)# epg EPG1`
	`ifav19-ifc1(config-tenant-app-epg)# show running-config`
	`# Command: show running-config`
	`tenant Test_Isolation`
	`application PVLAN epg EPG1`
	`tenant Test_Isolation`
	`application PVLAN`
	`epg EPG1`
	`bridge-domain member BD1`
	`contract consumer bare-metal`
	`contract consumer default`
	`contract provider Isolate_EPG`
	`isolation enforce <---- This enables EPG isolation mode.`
	`exit`
	`exit`
	`ifav19-ifc1(config)# leaf ifav19-leaf3`
	`ifav19-ifc1(config-leaf)# interface ethernet 1/16`
	`ifav19-ifc1(config-leaf-if)# show running-config`
	`ifav19-ifc1(config-leaf-if)# switchport trunk native vlan 101 tenant Test_Isolation application`	

	Command or Action	Purpose
	PVLAN epg StaticEPG primary-vlan 100 exit	
Step 2	Verify the configuration: **Example:** show epg StaticEPG detail Application EPg Data: Tenant : Test_Isolation Application : PVLAN AEPg : StaticEPG BD : BD1 uSeg EPG : no Intra EPG Isolation : **enforced** Vlan Domains : phys Consumed Contracts : bare-metal Provided Contracts : default,Isolate_EPG Denied Contracts : Qos Class : unspecified Tag List : VMM Domains: Domain Type Deployment Immediacy Resolution Immediacy State Encap Primary Encap ---------- ---------- ---------- ---------- ---------- ---------- DVS1 VMware On Demand immediate formed auto auto Static Leaves: Node Encap Deployment Immediacy Mode Modification Time ---------- ---------- ---------- ---------- ---------- Static Paths: Node Interface Encap Modification Time ---------- ---------- ---------- ---------- 1018 eth101/1/1 vlan-100 2016-02-11T18:39:02.337-08:00 1019 eth1/16 vlan-101 2016-02-11T18:39:02.337-08:00 Static Endpoints: Node Interface Encap End Point MAC End Point IP Address Modification Time ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ---------- ----------	

	Command or Action	Purpose

Intra-EPG Isolation for VMWare vDS

Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch

Intra-EPG Isolation is an option to prevent physical or virtual endpoint devices that are in the same base EPG or microsegmented (uSeg) EPG from communicating with each other. By default, endpoint devices included in the same EPG are allowed to communicate with one another. However, conditions exist in which total isolation of the endpoint devices from on another within an EPG is desirable. For example, you may want to enforce intra-EPG isolation if the endpoint VMs in the same EPG belong to multiple tenants, or to prevent the possible spread of a virus.

A Cisco Application Centric Infrastructure (ACI) virtual machine manager (VMM) domain creates an isolated PVLAN port group at the VMware VDS or Microsoft Hyper-V Virtual Switch for each EPG that has intra-EPG isolation enabled. A fabric administrator specifies primary encapsulation or the fabric dynamically specifies primary encapsulation at the time of EPG-to-VMM domain association. When the fabric administrator selects the VLAN-pri and VLAN-sec values statically, the VMM domain validates that the VLAN-pri and VLAN-sec are part of a static block in the domain pool.

Primary encapsulation is defined per EPG VLAN. In order to use primary encapsulation for Intra-EPG isolation, you must deploy it in one of the following ways:

- Segregate primary and secondary VLAN defined ports on different switches. EPG VLAN is created per switch. If you have port encapsulation, and only static ports on a switch for an EPG, primary encapsulation is not associated.
- Use a different encapsulation for static ports that use only port encapsulation. This creates a second EPG VLAN that does not have primary encapsulation associated with it.

  In the example below, consider egress traffic on two interfaces (Eth1/1, Eth1/3) with primary VLAN-1103. Eth1/1 port encap was changed to VLAN-1132 (from VLAN-1130), so that it does not share the secondary VLAN with Eth1/3.

```
Port encap with VLAN-1130 on Eth1/1
Eth1/1: Port Encap only VLAN-1130
Eth1/6: Primary VLAN-1103 and Secondary VLAN-1130


fab2-leaf3# show vlan id 53 ext

VLAN Name                              Encap            Ports
---- --------------------------------- ---------------- ------------------------
53   JT:jt-ap:EPG1-1                   vlan-1130        Eth1/1, Eth1/3

module-1# show sys int eltmc info vlan access_encap_vlan 1130
            vlan_id:             53   :::           isEpg:              1
         bd_vlan_id:             52   :::         hwEpgId:          11278
      srcpolicyincom:             0   :::       data_mode:              0
        accencaptype:             0   :::    fabencaptype:              2
       accencapval:           1130   :::     fabencapval:          12192
             sclass:          49154   :::         sglabel:             12
         sclassprio:              1   :::     floodmetptr:             13
         maclearnen:              1   :::       iplearnen:              1
        sclasslrnen:              1   :::   bypselffwdchk:              0
```

```
            qosusetc:              0   :::       qosuseexp:              0
            isolated:              1   :::   primary_encap:           1103
           proxy_arp:              0   :::       qinq core:              0
           ivxlan_dl:              0   :::       dtag_mode:              0
      is_service_epg:              0

Port encap changed to VLAN-1132 on Eth1/1
fab2-leaf3# show vlan id 62 ext

VLAN Name                             Encap            Ports
---- -------------------------------- ---------------- -------------------------
62   JT:jt-ap:EPG1-1                  vlan-1132        Eth1/1

module-1# show sys int eltmc info vlan access_encap_vlan 1132
          [SDK Info]:
             vlan_id:             62   :::           isEpg:             1
          bd_vlan_id:             52   :::         hwEpgId:         11289
      srcpolicyincom:              0   :::       data_mode:             0
        accencaptype:              0   :::    fabencaptype:             2
        accencapval:            1132   :::     fabencapval:         11224
              sclass:           49154   :::         sglabel:            12
          sclassprio:               1   :::     floodmetptr:            13
          maclearnen:               1   :::       iplearnen:             1
         sclasslrnen:               1   :::   bypselffwdchk:             0
            qosusetc:               0   :::       qosuseexp:             0
            isolated:               1   :::   primary_encap:             0
           proxy_arp:               0   :::       qinq core:             0
           ivxlan_dl:               0   :::       dtag_mode:             0
      is_service_epg:               0

fab2-leaf3# show vlan id 53 ext

VLAN Name                             Encap            Ports
---- -------------------------------- ---------------- -------------------------
53   JT:jt-ap:EPG1-1                  vlan-1130        Eth1/3

module-1# show sys int eltmc info vlan access_encap_vlan 1130
          [SDK Info]:
             vlan_id:             53   :::           isEpg:             1
          bd_vlan_id:             52   :::         hwEpgId:         11278
      srcpolicyincom:              0   :::       data_mode:             0
        accencaptype:              0   :::    fabencaptype:             2
        accencapval:            1130   :::     fabencapval:         12192
              sclass:           49154   :::         sglabel:            12
          sclassprio:               1   :::     floodmetptr:            13
          maclearnen:               1   :::       iplearnen:             1
         sclasslrnen:               1   :::   bypselffwdchk:             0
            qosusetc:               0   :::       qosuseexp:             0
            isolated:               1   :::   primary_encap:          1103
           proxy_arp:               0   :::       qinq core:             0
           ivxlan_dl:               0   :::       dtag_mode:             0
```

Note

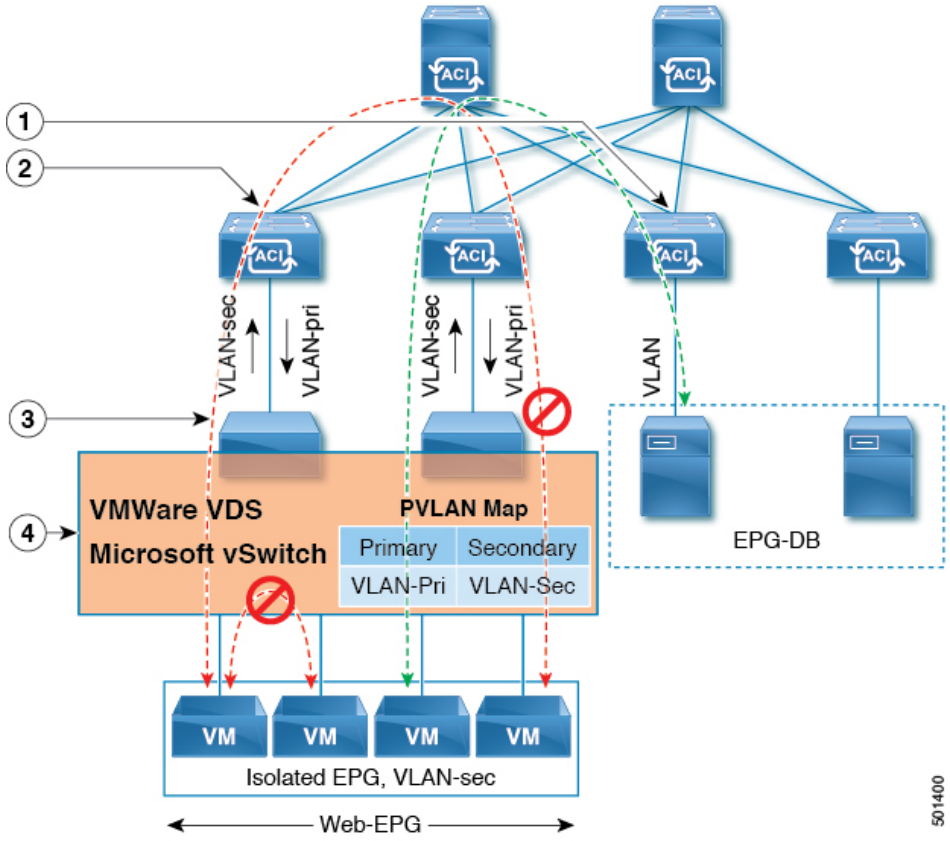
- When intra-EPG isolation is not enforced, the VLAN-pri value is ignored even if it is specified in the configuration.
- A VMware distributed virtual switch (DVS) domain with EDM UCSM integration may fail. The domain fails if you configure intra-EPG isolation on the endpoint group (EPG) attached to the domain and you use UCSM Mini 6324, which does not support private VLANs.

BPDUs are not forwarded through EPGs with intra-EPG isolation enabled. Therefore, when you connect an external Layer 2 network that runs spanning tree in a VLAN that maps to an isolated EPG on Cisco ACI, Cisco ACI might prevent spanning tree in the external network from detecting a Layer 2 loop. You can avoid this issue by ensuring that there is only a single logical link between Cisco ACI and the external network in these VLANs.

VLAN-pri/VLAN-sec pairs for the VMware VDS or Microsoft Hyper-V Virtual Switch are selected per VMM domain during the EPG-to-domain association. The port group created for the intra-EPG isolation EPGs uses the VLAN-sec tagged with type set to `PVLAN`. The VMware VDS or the Microsoft Hyper-V Virtual Switch and fabric swap the VLAN-pri/VLAN-sec encapsulation:

- Communication from the Cisco ACI fabric to the VMware VDS or Microsoft Hyper-V Virtual Switch uses VLAN-pri.
- Communication from the VMware VDS or Microsoft Hyper-V Virtual Switch to theCisco ACI fabric uses VLAN-sec.

Figure 15: Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch



Note these details regarding this illustration:

1. EPG-DB sends VLAN traffic to the Cisco ACI leaf switch. The Cisco ACI egress leaf switch encapsulates traffic with a primary VLAN (PVLAN) tag and forwards it to the Web-EPG endpoint.
2. The VMware VDS or Microsoft Hyper-V Virtual Switch sends traffic to the Cisco ACI leaf switch using VLAN-sec. The Cisco ACI leaf switch drops all intra-EPG traffic because isolation is enforced for all intra VLAN-sec traffic within the Web-EPG.

3. The VMware VDS or Microsoft Hyper-V Virtual Switch VLAN-sec uplink to the Cisco ACI leaf switch is in isolated trunk mode. The Cisco ACI leaf switch uses VLAN-pri for downlink traffic to the VMware VDS or Microsoft Hyper-V Virtual Switch.
4. The PVLAN map is configured in the VMware VDS or Microsoft Hyper-V Virtual Switch and Cisco ACI leaf switches. VM traffic from WEB-EPG is encapsulated in VLAN-sec. The VMware VDS or Microsoft Hyper-V Virtual Switch denies local intra-WEB EPG VM traffic according to the PVLAN tag. All intra-ESXi host or Microsoft Hyper-V host VM traffic is sent to the Cisco ACI leaf switch using VLAN-Sec.

Configuring Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch using the GUI

SUMMARY STEPS

1. Log into Cisco APIC.
2. Choose **Tenants** > ***tenant***.
3. In the left navigation pane expand the **Application Profiles** folder and appropriate application profile.
4. Right-click the **Application EPGs** folder and then choose **Create Application EPG**.
5. In the **Create Application EPG** dialog box, complete the following steps:
6. Click **Update** and click **Finish**.

DETAILED STEPS

Procedure

Step 1 Log into Cisco APIC.

Step 2 Choose **Tenants** > ***tenant***.

Step 3 In the left navigation pane expand the **Application Profiles** folder and appropriate application profile.

Step 4 Right-click the **Application EPGs** folder and then choose **Create Application EPG**.

Step 5 In the **Create Application EPG** dialog box, complete the following steps:

a) In the **Name** field, add the EPG name.
b) In the **Intra EPG Isolation** area, click **Enforced**.
c) In the **Bridge Domain** field, choose the bridge domain from the drop-down list.
d) Associate the EPG with a bare metal/physical domain interface or with a VM Domain.

- For the VM Domain case, check the **Associate to VM Domain Profiles** check box.
- For the bare metal case, check the **Statically Link with Leaves/Paths** check box.

e) Click **Next**.
f) In the **Associated VM Domain Profiles** area, click the **+** icon.
g) From the **Domain Profile** drop-down list, choose the desired VMM domain.

For the static case, in the **Port Encap (or Secondary VLAN for Micro-Seg)** field, specify the secondary VLAN, and in the **Primary VLAN for Micro-Seg** field, specify the primary VLAN. If the Encap fields are left blank, values will be allocated dynamically.

Note
For the static case, a static VLAN must be available in the VLAN pool.

Step 6 Click **Update** and click **Finish**.

Configuring Intra-EPG Isolation for VMware VDS or Microsoft Hyper-V Virtual Switch using the NX-OS Style CLI

SUMMARY STEPS

1. In the CLI, create an intra-EPG isolation EPG:
2. Verify the configuration:

DETAILED STEPS

Procedure

Step 1 In the CLI, create an intra-EPG isolation EPG:

Example:

The following example is for VMware VDS:

```
apic1(config)# tenant Test_Isolation
apic1(config-tenant)# application PVLAN
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# show running-config
# Command: show running-config tenant Tenant_VMM application Web epg intraEPGDeny
  tenant Tenant_VMM
    application Web
      epg intraEPGDeny
        bridge-domain member VMM_BD
        vmware-domain member PVLAN encap vlan-2001 primary-encap vlan-2002 push on-demand
        vmware-domain member mininet
          exit
        isolation enforce
        exit
      exit
    exit
apic1(config-tenant-app-epg)#
```

Example:

The following example is for Microsoft Hyper-V Virtual Switch:

```
apic1(config)# tenant Test_Isolation
apic1(config-tenant)# application PVLAN
apic1(config-tenant-app)# epg EPG1
apic1(config-tenant-app-epg)# show running-config
# Command: show running-config tenant Tenant_VMM application Web epg intraEPGDeny
  tenant Tenant_VMM
    application Web
      epg intraEPGDeny
        bridge-domain member VMM_BD
        microsoft-domain member domain1 encap vlan-2003 primary-encap vlan-2004
        microsoft-domain member domain2
```

```
          exit
        isolation enforce
        exit
      exit
    exit
apic1(config-tenant-app-epg)#
```

Step 2 Verify the configuration:

Example:

```
 show epg StaticEPG detail
Application EPg Data:
Tenant              : Test_Isolation
Application         : PVLAN
AEPg                : StaticEPG
BD                  : VMM_BD
uSeg EPG            : no
Intra EPG Isolation : enforced
Vlan Domains        : VMM
Consumed Contracts  : VMware_vDS-Ext
Provided Contracts  : default,Isolate_EPG
Denied Contracts    :
Qos Class           : unspecified
Tag List            :
VMM Domains:
Domain                Type       Deployment Immediacy  Resolution Immediacy  State           Encap
      Primary
Encap
 --------------------  ---------  --------------------  --------------------  --------------  ----------
  ----------
 DVS1                  VMware     On Demand             immediate             formed          auto
       auto

Static Leaves:
 Node        Encap             Deployment Immediacy  Mode                Modification Time

----------  ----------------  --------------------  ------------------  ------------------------------


Static Paths:
 Node        Interface                       Encap             Modification Time
 ----------  ------------------------------  ----------------  -----------------------------
 1018        eth101/1/1                      vlan-100          2016-02-11T18:39:02.337-08:00
 1019        eth1/16                         vlan-101          2016-02-11T18:39:02.337-08:00

Static Endpoints:
 Node        Interface          Encap             End Point MAC      End Point IP Address
Modification Time
 ----------  ------------------------------  ----------------  -----------------
------------------------------  ------------------------------

Dynamic Endpoints:
Encap: (P):Primary VLAN, (S):Secondary VLAN
 Node        Interface            Encap             End Point MAC      End Point IP Address
 Modification Time
 ----------  ------------------------------  ----------------  -----------------
------------------------------  ------------------------------
1017        eth1/3              vlan-943(P)       00:50:56:B3:64:C4  ---
2016-02-17T18:35:32.224-08:00
                                vlan-944(S)
```

Configuring Intra-EPG Isolation for Cisco ACI Virtual Edge

Intra-EPG Isolation Enforcement for Cisco ACI Virtual Edge

By default, endpoints with an EPG can communicate with each other without any contracts in place. However, you can isolate endpoints within an EPG from each other. For example, you may want to enforce endpoint isolation within an EPG to prevent a VM with a virus or other problem from affecting other VMs in the EPG.

You can configure isolation on all or none of the endpoints within an application EPG; you cannot configure isolation on some endpoints but not on others.

Isolating endpoints within an EPG does not affect any contracts that enable the endpoints to communicate with endpoints in another EPG.

Note Enforcing intra-EPG Isolation is not supported for the EPG that is associated with Cisco ACI Virtual Edge domains in VLAN mode. If you try to enforce intra-EPG isolation with such an EPG, a fault is triggered.

Note Using intra-EPG isolation on a Cisco ACI Virtual Edge microsegment (uSeg) EPG is not currently supported.

Note Proxy ARP is not supported for Cisco ACI Virtual Edge EPGs using VXLAN encapsulation and on which intra-EPG Isolation is enforced. Therefore, intra-subnet communication is not possible between intra-EPG isolated EPGs even though contracts are in place between those Cisco ACI Virtual Edge EPGs. (VXLAN).

Configure Intra-EPG Isolation for Cisco ACI Virtual Edge Using the GUI

Follow this procedure to create an EPG in which the endpoints of the EPG are isolated from each other.

The port that the EPG uses must belong to one of the VM Managers (VMMs).

Note This procedure assumes that you want to isolate endpoints within an EPG when you create the EPG. If you want to isolate endpoints within an existing EPG, select the EPG in Cisco APIC, and in the **Properties** pane, in the **Intra EPG Isolation** area, choose **Enforced**, and then click **SUBMIT**.

Before you begin

Make sure that VXLAN-related configuration is present on the Cisco ACI Virtual Edge VMM domain, particularly a Cisco ACI Virtual Edge fabric-wide multicast address and pool of multicast addresses (one per EPG).

Procedure

Step 1 Log in to Cisco APIC.

Step 2 Choose **Tenants**, expand the folder for the tenant, and then expand the **Application Profiles** folder.

Step 3 Right-click an application profile, and choose **Create Application EPG**.

Step 4 In the **Create Application EPG** dialog box, complete the following steps:

a) In the **Name** field, enter the EPG name.

b) In the **Intra EPG Isolation** area, click **Enforced**.

c) From the **Bridge Domain** drop-down list, choose the bridge domain.

d) Check the **Associate to VM Domain Profiles** check box.

e) Click **Next**.

f) In the **Associate VM Domain Profiles** area, complete the following steps:

- Click the + (plus) icon, and from the **Domain Profile** drop-down list, choose the desired Cisco ACI Virtual Edge VMM domain.
- From the **Switching Mode** drop-down list, choose **AVE**.
- From the **Encap Mode** drop-down list, choose **VXLAN** or **Auto**.

  If you choose **Auto**, make sure that encapsulation mode of the Cisco ACI Virtual Edge VMM domain is VXLAN.
- (Optional) Choose other configuration options appropriate to your setup.

g) Click **Update** and click **Finish**.

What to do next

You can select statistics and view them to help diagnose problems involving the endpoint. See the sections Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab, on page 61 and View Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab, on page 62 in this guide.

Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab

If you configured intra-EPG isolation on a Cisco ACI Virtual Edge, choose statistics—such as denied connections, received packets, or transmitted multicast packets—for the endpoints. You can then view the statistics.

Procedure

Step 1 Log in to Cisco APIC.

Step 2 Choose **Tenants** > ***tenant***.

Step 3 In the tenant navigation pane, expand the **Application Profiles**, ***profile***, and **Application EPGs** folders, and then choose the EPG containing the endpoint the statistics for which you want to view.

Step 4 In the EPG **Properties** work pane, click the **Operational** tab to display the endpoints in the EPG.

Step 5 Double-click the endpoint.

Step 6 In the **Properties** dialog box for the endpoint, click the **Stats** tab and then click the check icon.

Step 7 In the **Select Stats** dialog box, in the **Available** pane, choose the statistics that you want to view for the endpoint, and then use the right-pointing arrow to move them into the **Selected** pane.

Step 8 Click **Submit**.

View Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab

If you configured intra-EPG isolation on a Cisco ACI Virtual Edge, once you have chosen statistics for the endpoints, you can view them.

Before you begin

You must have chosen statistics to view for isolated endpoints. See Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab, on page 61 in this guide for instructions.

Procedure

Step 1 Log in to Cisco APIC.

Step 2 Choose **Tenants** > ***tenant***.

Step 3 In the tenant navigation pane, expand the **Application Profiles**, ***profile***, and **Application EPGs** folders, and then choose the EPG containing the endpoint with statistics that you want to view.

Step 4 In the EPG **Properties** work pane, click the **Operational** tab to display the endpoints in the EPG.

Step 5 Double-click the endpoint with statistics that you want to view.

Step 6 In the **Properties** work pane for the endpoint, click the **Stats** tab.

The work pane displays the statistics that you chose earlier. You can change the view by clicking the table view or chart view icon on the upper left side of the work pane.

Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Virtual Networking Tab

If you configured intra-EPG isolation on a Cisco ACI Virtual Edge, choose statistics—such as denied connections, received packets, or transmitted multicast packets—for the endpoints. You can then view the statistics.

Procedure

Step 1 Log in to Cisco APIC.

Step 2 Choose **Virtual Networking** > **Inventory** > **VMM Domains** > **VMware** > ***VMM domain*** > **Controllers** > ***controller instance name*** > ***DVS-VMM name*** > **Portgroups** > ***EPG name*** > ***Learned Point MAC address (node)*** > **.**

Step 3 Click the **Stats** tab.

Step 4 Click the tab with the check mark.

Step 5 In the **Select Stats** dialog box, click the statistics that you want to view in the **Available** pane, and then click the arrow pointing right to put them in the **Selected** pane.

Step 6 (Optional) Choose a sampling interval.

Step 7 Click **Submit**.

View Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Virtual Networking Tab

If you configured intra-EPG isolation on a Cisco ACI Virtual Edge, once you have chosen statistics for the endpoints, you can view them.

Before you begin

You must have chosen statistics to view for isolated endpoints. See Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab, on page 61 in this guide for instructions.

Procedure

Step 1 Log in to Cisco APIC.

Step 2 Choose **Virtual Networking** > **Inventory** > **VMM Domains** > **VMware** > ***VMM name*** > **Controllers** > **controller instance name** > ***DVS-VMM name*** > **Portgroups** > ***EPG name*** > ***Learned Point MAC address (node)***

Step 3 Click the **Stats** tab.

The central pane displays the statistics that you chose earlier. You can change the view by clicking the table view or chart view icon on the upper left side of the work pane.

Configure Intra-EPG Isolation for Cisco ACI Virtual Edge Using the NX-OS Style CLI

Before you begin

Make sure that VXLAN-related configuration is present on the Cisco ACI Virtual Edge VMM domain, particularly a Cisco ACI Virtual Edge fabric-wide multicast address and pool of multicast addresses (one per EPG).

Procedure

In the CLI, create an intra-EPG isolation EPG:

Example:

```
# Command: show running-config tenant Tenant2 application AP-1 epg EPG-61
  tenant Tenant2
    application AP-1
      epg EPG-61
        bridge-domain member BD-61
        vmware-domain member D-AVE-SITE-2-3
          switching-mode AVE
          encap-mode vxlan
          exit
        isolation enforce            # This enables EPG into isolation mode.
        exit
      exit
    exit
```

What to do next

You can select statistics and view them to help diagnose problems involving the endpoint. See the sections Choose Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab, on page 61 and View Statistics for Isolated Endpoints on Cisco ACI Virtual Edge Under the Tenants Tab, on page 62 in this guide.

Troubleshooting

Troubleshooting Endpoint Connectivity

Procedure

Step 1 Inspect the operational status of each endpoint.

The operational status will reveal any fault or misconfiguration of the endpoints. See

Inspecting the Endpoint Status, on page 65 .

Step 2 Inspect the status of the tunnel interface.

The operational status will reveal any fault or misconfiguration of the tunnel. See Inspecting the Tunnel Interface Status, on page 66.

Step 3 Perform a traceroute between the endpoint groups (EPGs).

A traceroute will reveal any problems with intermediate nodes, such as spine nodes, between the endpoints. See Performing a Traceroute Between Endpoints, on page 66.

Step 4 Configure an atomic counter on an endpoint.

The atomic counter will confirm whether the source endpoint is transmitting packets or the destination endpoint is receiving packets, and whether the number of packets received equals the number of packets sent. See Configuring Atomic Counters, on page 67.

Step 5 Inspect the contracts under each EPG.

Inspect the contracts under each EPG to make sure they allow the traffic that should flow between the EPGs. As a test, you can temporarily open the contracts to allow unrestricted traffic.

Step 6 Configure a SPAN policy to forward source packets to a monitoring node.

A packet analyzer on the monitoring node will reveal any packet issues such as an incorrect address or protocol. See Configuring a Tenant SPAN Session Using the Cisco APIC GUI, on page 68.

Inspecting the Endpoint Status

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the tenant that contains the source endpoint.

Step 3 In the **Navigation** pane, expand the tenant, expand **Application Profiles**, and expand the application profile that contains the endpoint.

Step 4 Expand **Application EPGs** and click the EPG to be inspected.

Step 5 In the **Work** pane, from the list of endpoints in the **Endpoint** table, double-click the source endpoint to open the **Client End Point** dialog box.

Step 6 In the **Client End Point** dialog box, verify the endpoint properties and click the **Operational** tab.

Step 7 In the **Operational** tab, view the health, status, and fault information.

In the **Status** table, click any items with entries, such as changes, events, or faults.

Step 8 Close the **Client End Point** dialog box.

Step 9 In the **Endpoint** table, view the **Interface** entry for the endpoint and note the node and tunnel IDs.

Step 10 Repeat this procedure for the destination endpoint.

Note
Occasionally, bidirectional traffic is interrupted between IP addresses in two micro-segmented EPGs deployed behind two leaf switches in the fabric. This can occur when the IP addresses are transitioning because of a configuration change from micro-segment EPG to base EPG. Or conversely, this can occur on two different leaf switches at the same time

while bidirectional traffic is running. In this case, the policy tag for each remote endpoint still points to its previous EPG.

Workaround: Manually clear the remote endpoints on the switches or wait for the remote endpoint to age out. To clear the endpoints, log on to the CLI on each switch and enter the **clear system internal epm endpoint** command with the appropriate option. For example, if your endpoints are based on the IP address, enter **clear system internal epm endpoint key vrf** *vrf_name***{ip | ipv6}** *ip-address*. The endpoints are then relearned with the correct policy tag.

Inspecting the Tunnel Interface Status

This procedure shows how to inspect the operational status of the tunnel interface.

Procedure

Step 1 In the menu bar, click **Fabric**.

Step 2 In the submenu bar, click **Inventory**.

Step 3 In the **Navigation** pane, expand the pod and expand the node ID of the source endpoint interface.

Step 4 Under the node, expand **Interfaces**, expand **Tunnel Interfaces**, and click the tunnel ID of the source endpoint interface.

Step 5 In the **Work** pane, verify the tunnel interface properties and click the **Operational** tab.

Step 6 In the **Operational** tab, view the health, status, and fault information.

In the **Status** table, click any items with entries, such as changes, events, or faults.

Step 7 Repeat this procedure for the destination endpoint interface.

Performing a Traceroute Between Endpoints

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the tenant that contains the source endpoint.

Step 3 In the **Navigation** pane, expand the tenant and expand **Policies** > **Troubleshoot**.

Step 4 Under **Troubleshoot**, right-click on one of the following traceroute policies:

- **Endpoint-to-Endpoint Traceroute Policies** and choose **Create Endpoint-to-Endpoint Traceroute Policy**
- **Endpoint-to-External-IP Traceroute Policies** and choose **Create Endpoint-to-External-IP Traceroute Policy**
- **External-IP-to-Endpoint Traceroute Policies** and choose **Create External-IP-to-Endpoint Traceroute Policy**
- **External-IP-to-External-IP Traceroute Policies** and choose **Create External-IP-to-External-IP Traceroute Policy**

Step 5 Enter the appropriate values in the dialog box fields and click **Submit**.

Note
For the description of a field, click the help icon (**?**) in the top-right corner of the dialog box.

Step 6 In the **Navigation** pane or the **Traceroute Policies** table, click the traceroute policy.
The traceroute policy is displayed in the **Work** pane.

Step 7 In the **Work** pane, click the **Operational** tab, click the **Source Endpoints** tab, and click the **Results** tab.

Step 8 In the **Traceroute Results** table, verify the path or paths that were used in the trace.

Note
- More than one path might have been traversed from the source node to the destination node.
- For readability, you can increase the width of one or more columns, such as the **Name** column.

Configuring Atomic Counters

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the desired tenant.

Step 3 In the **Navigation** pane, expand the tenant and expand **Policies** and then expand **Troubleshoot**.

Step 4 Under **Troubleshoot**, expand **Atomic Counter Policy** and choose a traffic topology.
You can measure traffic between a combination of endpoints, endpoint groups, external interfaces, and IP addresses.

Step 5 Right-click the desired topology and choose **Add** *topology* **Policy** to open an **Add Policy** dialog box.

Step 6 In the **Add Policy** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the policy.

b) choose or enter the identifying information for the traffic source.
The required identifying information differs depending on the type of source (endpoint, endpoint group, external interface, or IP address).

c) choose or enter the identifying information for the traffic destination.

d) (Optional) (Optional) In the **Filters** table, click the **+** icon to specify filtering of the traffic to be counted.
In the resulting **Create Atomic Counter Filter** dialog box, you can specify filtering by the IP protocol number (TCP=6, for example) and by source and destination IP port numbers.

e) Click **Submit** to save the atomic counter policy.

Step 7 In the **Navigation** pane, under the selected topology, choose the new atomic counter policy.
The policy configuration is displayed in the **Work** pane.

Step 8 In the **Work** pane, click the **Operational** tab and click the **Traffic** subtab to view the atomic counter statistics.

Configuring a Tenant SPAN Session Using the Cisco APIC GUI

SPAN can be configured on a switch or on a tenant. This section guides you through the Cisco APIC GUI to configure a SPAN policy on a tenant to forward replicated source packets to a remote traffic analyzer. The configuration procedure requires entering values in the fields of one or more GUI dialog boxes. To understand a field and determine a valid value, view the help file by clicking the help icon (**?**) at the top-right corner of the dialog box.

Procedure

Step 1 In the menu bar, click **Tenants**.

Step 2 In the submenu bar, click the tenant that contains the source endpoint.

Step 3 In the **Navigation** pane, expand the tenant, expand **Policies** > **Troubleshooting** > **SPAN**.

Two nodes appear under **SPAN**: **SPAN Destination Groups** and **SPAN Source Groups**.

Step 4 From the **Navigation** pane, right-click **SPAN Source Groups** and choose **Create SPAN Source Group**.
The **Create SPAN Source Group** dialog appears.

Step 5 Enter the appropriate values in the required fields of the **Create SPAN Source Group** dialog box.

Step 6 Expand the **Create Sources** table to open the **Create SPAN Source** dialog box.

Step 7 Enter the appropriate values in the **Create SPAN Source** dialog box fields.

Step 8 When finished creating the SPAN source, click **OK**.

You return to the **Create SPAN Source Group** dialog box.

Step 9 When finished entering values in the **Create SPAN Source Group** dialog box fields, click **Submit**.

What to do next

Using a traffic analyzer at the SPAN destination, you can observe the data packets from the SPAN source EPG to verify the packet format, addresses, protocols, and other information.

Verifying IP-Based EPG Configurations

There are two types of endpoint groups (EPGs) that you can create: application EPGs and IP-based EPGs. IP-based EPGs differ from regular application EPGs in that they are microsegment EPGs. This chapter explains how to verify that your IP-based EPG configurations are properly classified as IP-based using the GUI or using switch commands.

This chapter contains the following sections:

Verifying IP-Based EPG Configurations Using the GUI

This procedure explains how to verify that you have correctly configured an IP-based EPG using the GUI and Visore tool.

Procedure

Step 1 Verify that the IP-based EPG you created is listed under the **uSeg EPGs** folder in the GUI (shown in the following screen capture).
Note that there is one IP-based EPG listed under uSeg EPGs named "IP" that was created using the REST API.

Step 2 Verify that the information is correct in the EPG - IP properties screen (right side window pane) for each EPG IP (IP-based EPG).
Note the list of IP-based EPGs and IP addresses that are shown at the bottom of the screen.

Step 3 From your web browser, enter the APIC IP address followed by "/visore.html." Visore is a tool that allows you to view all the objects in the system, such as EPGs. You can use Visore to verify that your IP-based EPGs have been properly configured. For more information about Visore, see the *Application Policy Infrastructure Controller Visore Tool Introduction* document.

Step 4 Enter your username and password then click **Login** to log into Visore.

Step 5 Run a query for the IP-based EPGs that you verified in the GUI by entering the name of the class in the field next to **Class or DN** (for example, "fvAEPg").

Note
This is a view from the APIC point of view. You can see that the "Total objects shown" above is "3", meaning there are three EPGs that were downloaded to the switch. You can see that the IP-based EPG that was previously listed in the GUI as "IP" is now shown next to "dn". Also note that "yes" is displayed next to "isAttrBasedEPg", which means that this has been properly configured as an IP-based EPG. You can verify all the objects have been configured successfully using Visore, including both application EPGs and IP-based EPGs.

Step 6 This is a view from the switch point of view. On the switch, you can run a query for the fvEpP class to see the EPGs and check for the "crtrnEnabled" attribute. It will be set to "yes" for IP-based EPGs.
Verify that under this EPG, the children of the EPG are shown with IP addresses to ensure a proper configuration. For each IP address configured, there is one object (named "l3IpCktEp") that the switch uses to classify the traffic. Once the configuration is there, when the packets arrive, the switch uses these objects to classify them.

Step 7 Verify that the pcTags for all the endpoints and IP addresses that you configured match. Every EPG has a pcTag. All the endpoints that match with the IP addresses you configured are classified into this pcTag. Every endpoint has an IP address that you can run a class query on. When you are troubleshooting, you want to verify whether these endpoints (servers) are properly getting classified into this IP-based EPG or not. (The pcTags should match for the IP-based EPG.)

Verifying IP-EPG Configurations Using Switch Commands

This procedure explains how to use switch commands to verify you IP-EPG ("IpCkt") configurations.

Procedure

Step 1 Log in to the leaf.

Step 2 Navigate to the /mit/sys directory.

Step 3 In the /mit/sys directory, find ctx (vrf context directory)

Step 4 In the VRF cts directory, go to the specific BD directory where the IpCkt is configured.

You should see the IpCkt.

Note
"IpCkt" and "IP-EPG" are used interchangeably in this document.

Step 5 Navigate to the directory and the "cat summary" gives you the information regarding IpCkt.

Step 6 Ensure that the summary's "operSt' does not say "unsupported".

Step 7 Find out the VLAN ID that corresponds to the BD where the IpCkt is configured.

Note
The VLAN ID can be found through any of the **show vlan internal bd-info** commands or through the **show system internal epm vlan all** command.

Step 8 Once you find the VLAN ID of the BD, issue **show system internal epm <vlan-id> detail**.
Here you should be able to see all the configured IpCkts with a specific sclass. (It should match that of what you see in the /mit/sys directory.)

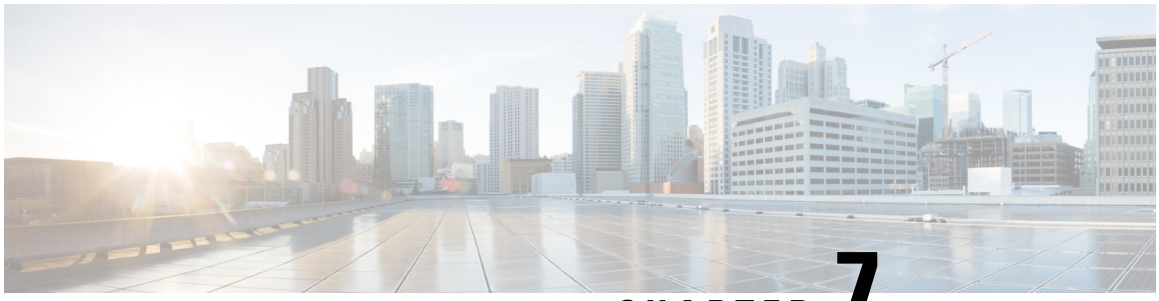
Step 9 Repeat the steps for vsh_lc that you followed for vsh.

Step 10 Send the traffic with an IP matching the IpCtk in the BD, and through **show system internal epm endp ip <a.b.c.d>**, you can verify that the learned IP has the IP-flags for "sclass" and a specific sclass value.

Step 11 Repeat the steps for vsh_lc that you followed for vsh.

List of the Switch Troubleshooting Commands Used in this Procedure:

```
Cd /mits/sys/ctx-vxlan…/bd-vxlan…
     - cat summary
Vsh -c "show system internal epm vlan all" or
Vsh -c "show vlan internal bd-info"
Vsh -c "show system internal epm vlan <vlan-id> detail"
Vsh -c "show system internal epm endp ip <a.b.c.d>"
Vsh_lc -c "show system internal epm vlan all" or
Vsh_lc -c "show vlan internal bd-info"
Vsh_lc -c "show system internal epm vlan <vlan-id> detail"
vsh_lc -c "show system internal epm endp ip <a.b.c.d>"
vsh_lc -c "show system internal epm epg"
```



CHAPTER 7

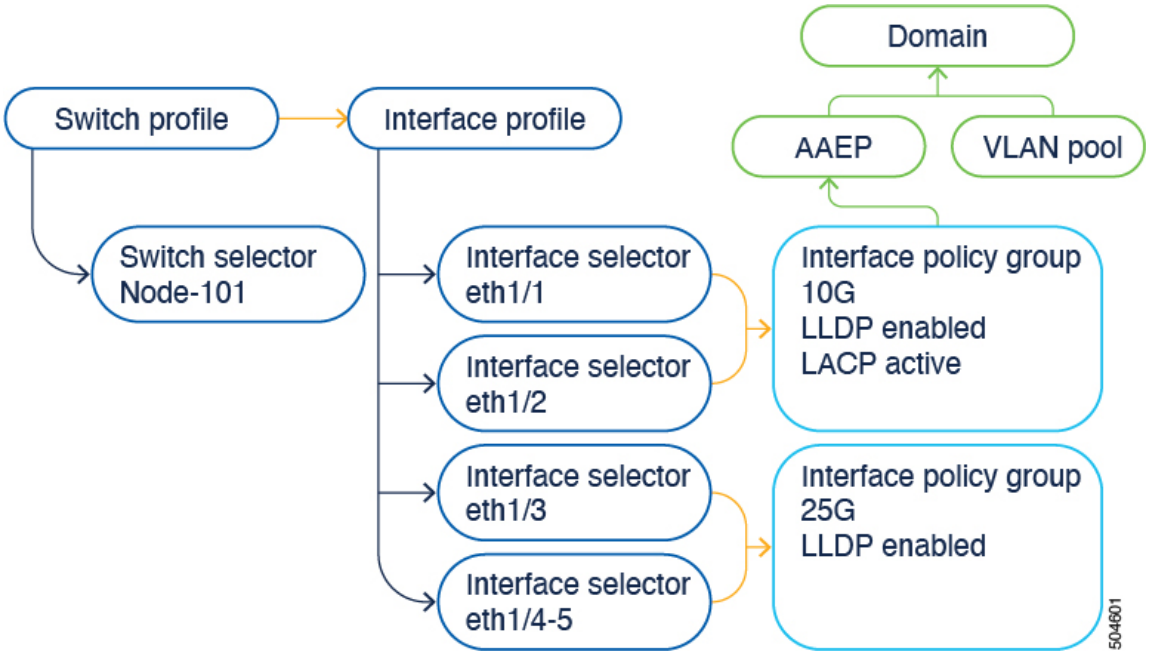
Access Interfaces

- About Access Interfaces, on page 71
- Physical Ports Configuration, on page 74
- Port Channels, on page 81
- Virtual Port Channels in Cisco ACI, on page 89
- Reflective Relay (802.1Qbg), on page 103
- Configuring Port, PC, and vPC Connections to FEX Devices, on page 105
- Configuring Port Profiles, on page 110
- Editing an Interface Configuration, on page 122

About Access Interfaces

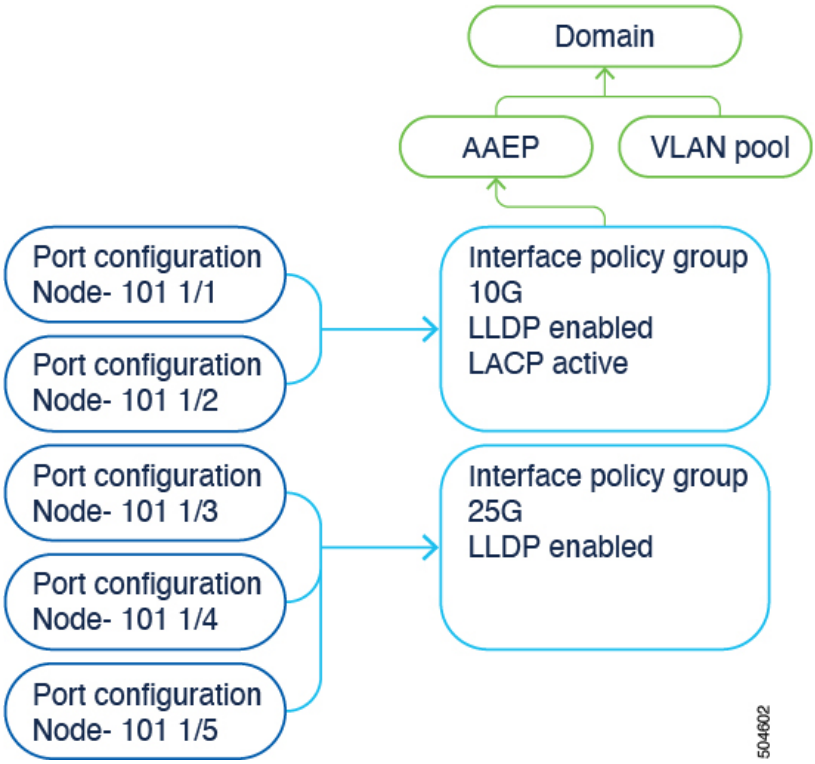
In Cisco Application Centric Infrastructure (ACI), interface configurations are performed by associating an interface policy group, which is a group of interface policies such as interface speed or link layer discovery protocol (LLDP), to an interface on a switch node. Cisco ACI uses four objects (switch profile, switch selector, interface profile, and interface selector) to select a certain interface on a certain switch node. This document refers to this mode of operations as the "profiles and selectors configuration." The following figure illustrates this configuration:

Figure 16: Interface configuration based on profiles and selectors



The Cisco ACI 6.0(1) release adds the "per-port configuration" configuration option (also known as the "interface configuration" or `infraPortConfig`, which is the name of the object for this configuration) that simplifies the interface configuration. This option presents the four objects as a single object and has the object specify an interface on a switch node. As a result, you do not need to use nor maintain switch profiles, switch selectors, interface profiles, and interface selectors.

Figure 17: Per-port interface configuration



You can access the per-port configuration in the following ways in the Cisco Application Policy Infrastructure Controller (APIC) GUI:

- **Fabric** > **Access Policies** > **Interface Configurations**
- **Fabric** > **Access Policies** > **Quick Start** > **Configure Interfaces**
- **Fabric** > **Inventory** > ***pod_ID*** > ***switch_name*** > **Interface tab** > **Configure Interfaces**

You can configure a switch using switch profiles and selectors and interface profiles and selectors at the same locations as before in the Cisco APIC GUI:

- **Fabric** > **Access Policies** > **Switches** > **Leaf Switches** > **Profiles**
- **Fabric** > **Access Policies** > **Switches** > **Spine Switches** > **Profiles**
- **Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Profiles**
- **Fabric** > **Access Policies** > **Interfaces** > **Spine Interfaces** > **Profiles**

However, we recommend that you use the per-port configuration.

When using the interface configuration option, the Cisco APIC creates and maintains switch profiles and selectors and interface profiles and selectors as read-only with as few objects as possible. For example, if you configure two contiguous ports identically, the Cisco APIC automatically creates a range in the configuration. You configure the ports individually and you do not have to worry about these optimizations; the Cisco APIC takes care of them. These objects that the Cisco APIC creates automatically are called "system-generated profiles" and you do not need to maintain them.

The system-generated profiles are still visible under **Fabric** > **Access Policies** > **Interfaces** > **{Leaf | Spine} {Switches | Interfaces}** > **Profiles** in the GUI along with any user-defined profiles.

If you configure an interface using the interface configuration option and you previously configured the interface with profiles and selectors, the Cisco APIC automatically removes the interface from the existing profiles and moves the interface to the system-generated profiles seamlessly. If the pre-existing switch and interface profiles contain other interfaces, the Cisco APIC does not delete them; you can keep using them in the traditional way. If the pre-existing profiles no longer contain any interfaces, the Cisco APIC automatically removes those profiles because they are no longer needed.

If you already configured an interface using a multinode selector, meaning that you assigned the port selector to a profile with multiple leaf switches, you must simultaneously configure the same interface for each node that belongs to the multinode selector for the Cisco APIC to remove those nodes automatically from the existing profile. Otherwise, a validation failure blocks the migration.

Physical Ports Configuration

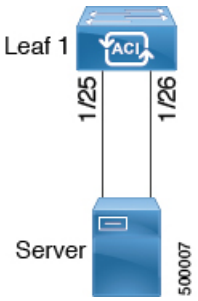
There are multiple ways to configure Cisco Application Centric Infrastructure (ACI) leaf switch interfaces:

- By using the selector and profile-based configuration model. From **Fabric** > **Access Policies** > **Switches** > **Leaf Switches** > **Profiles**, you can configure a switch profile that selects a leaf node with a leaf selector and the associated interface profiles to select the interface profiles (**Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Profiles**), which in turn select one or more interfaces and associate them to interface policy groups.
- By using the interface configuration beginning with the Cisco Application Policy Infrastructure Controller (APIC) 6.0(1) release. Go to **Fabric** > **Access Policies** > **Interface Configurations**. This configuration option simplifies the configuration workflow by reducing the number of configuration steps from four to one.
- By using the inventory view from **Fabric** > **Inventory** > ***pod_ID*** > ***switch_name***. Beginning with the Cisco APIC 6.0(1) release, the inventory view configuration also uses the interface configuration.
- By using the **Fabric** > **Access Policies** > **Quick Start** wizard. Beginning with the Cisco APIC 6.0(1) release, the inventory view configuration also uses the interface configuration.

Configuring Leaf Switch Physical Ports Using the Interface Configuration Model Using the GUI

This procedure uses either the **Fabric** > **Access Policies** > **Quick Start** > **Configure Interfaces** or the **Fabric** > **Access Policies** > **Interface Configuration** page to attach a server to a Cisco Application Centric Infrastructure (ACI) leaf switch interface. The steps would be the same for attaching other kinds of devices to a Cisco ACI leaf switch interface.

Figure 18: Switch Interface Configuration for Bare Metal Server



Before you begin

- The Cisco ACI fabric is installed, Cisco Application Policy Infrastructure Controllers (APICs) are online, and the Cisco APIC cluster is formed and healthy.
- A Cisco APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the Cisco ACI fabric and available.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the **Navigation** pane, choose **Quick Start** or **Interface Configuration**.

Step 3 In the **Work** pane, click **Configure Interfaces**. of the Quick Start wizard, click **Configure Interfaces**, or in the **Work** pane of **Interface Configuration** choose **Actions** > **Configure Interfaces**.

Step 4 In the **Configure Interfaces** dialog, perform the following actions:

a) For **Node Type**, click **Leaf**.
b) For **Port Type**, click **Access**.
c) For **Interface Type**, choose the desired type.
d) For **Interface Aggregation Type**, choose **Individual**.
e) For **Node**, click **Select Node**, put a check in the box for the desired switch (node), then click **OK**. You can select multiple switches.
f) For **Interfaces For All Switches**, enter the range of desired interfaces.
g) For **Leaf Access Port Policy Group**, click **Select Leaf Access Port Policy Group**.
h) In the **Select Leaf Access Port Policy Group** dialog, click **Create Leaf Access Port Policy Group**.

The interface policy group is a named policy that specifies the group of interface policies you will apply to the selected interfaces of the switch. Examples of interface policies include link level policy (for example, 1gbit port speed) and storm control interface policy.

i) In the **Create Leaf Access Port Policy Group** dialog, choose or create the desired policies.
j) Click **Save**.

What to do next

This completes the basic leaf switch interface configuration steps.

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Migrating Interfaces From the Selector and Profile to Interface Configuration Using the GUI

You can convert the configuration of an existing interface from the selector- and profile-based model to the interface configuration model by using this procedure.

Note The Cisco Application Policy Infrastructure Controller (APIC) does not automatically migrate interfaces with an active policy group override. You must migrate these ports manually.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the **Navigation** pane, choose **Interface Configuration**.

Step 3 In the table, select the interface that you want to migrate and click the three dots at the right.

Step 4 In the pop-up menu, choose **Edit Interface Configuration**.

The following a message displays:

```
This interface is configured using interface selectors. We recommend migrating it to newer way of
configuring interfaces. Clicking on Save will migrate this interface.
```

Step 5 Click **Save**.

The Cisco APIC converts the interface to the new configuration model.

Step 6 Perform one of the following sets of substeps depending on your Cisco APIC release and what you want to do:

To migrate a single interface:

a) In the table, select the interface that you want to migrate and click the three dots at the right.

b) In the pop-up menu, choose **Edit Interface Configuration**.

The following a message displays:

```
This interface is configured using interface selectors. We recommend migrating it to newer way of
configuring interfaces. Clicking on Save will migrate this interface.
```

c) Click **Save**.

The Cisco APIC converts the interface to the new configuration model.

In the 6.0(2) release and later, Cisco APIC simplifies the task of migrating existing configurations based on the selector- and profile-based model to the interface configuration model. You can migrate the selector-based configuration for all the ports of a node by selecting multiple nodes. This capability is useful if a selector spans across multiple nodes. To migrate multiple interfaces:

a) In the table, select the interfaces that you want to migrate.
b) Choose **Actions** > **Configure Interfaces**.

   The following a message displays:

   ```
   This interface is configured using interface selectors. We recommend migrating it to newer way of configuring interfaces. Clicking on Save will migrate this interface.
   ```

c) Click **Save**.

   The Cisco APIC converts the interfaces to the new configuration model.

Modifying the Interface Configuration Using the GUI

You can modify the configuration of an interface as follows:

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.
Step 2 In the **Navigation** pane, choose **Interface Configuration**.
Step 3 In the table, select the interface that you want to migrate and click the three dots at the right.
Step 4 In the pop-up menu, choose **Edit Interface Configuration**.

A window appears with the policy group associated to this interface.

Step 5 If there is an existing policy group, you can remove it by clicking the **x** next to the group.
Step 6 Click **Select Leaf Access Port Policy Group** to assign a new policy group.
Step 7 Choose an existing policy group or click **Create Leaf Access Port Policy Group** to create a new one.
Step 8 Click **Save**.

Viewing the Interface Configuration Using the GUI

The Cisco Application Policy Infrastructure Controller (APIC) GUI offers a unified view of the interface configuration regardless of whether the interfaces have been configured using the selector and profile model or the interface configuration model.

Choose **Fabric** > **Access Policies** > **Interface Configuration** and navigate the table on the right to see all the leaf nodes and interfaces.

Click a leaf node to view the leaf node's information, such as the admin state, the IP address of the TEP, the ID number, the hardware model, the serial number, and the software version.

Click an interface to view the interface's information. This view is called the "Infra Port Summary." Click the middle icon on the top right for a full screen view of the interface's information. The full screen view contains the following tabs that show additional information: **Overview**, **Operational**, **Deployed EPGs**, **VLANs**, **Statistics**, **QoS stats**, and **Event Analytics**. Click the **x** in the top right to close this full screen view.

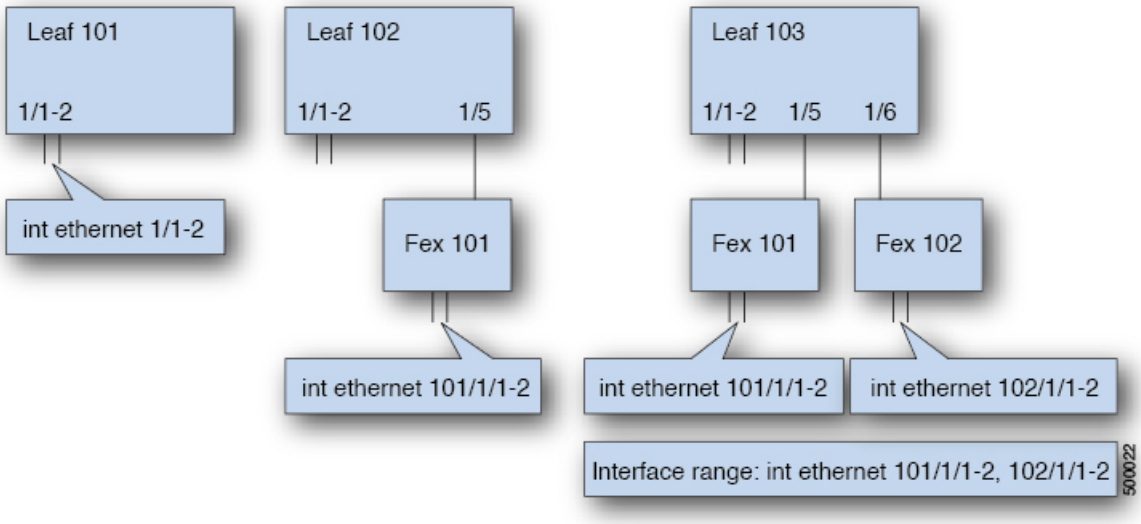
Click the policy group name of a given interface to view information about the policy group, such as the 802.1X configuration, attachable entity profile, CDP configuration, and LLDP configuration.

Configuring Physical Ports in Leaf Nodes and FEX Devices Using the NX-OS CLI

The commands in the following examples create many managed objects in the Cisco Application Centric Infrastructure (ACI) policy model that are fully compatible with the REST API/SDK and GUI. However, the CLI user can focus on the intended network configuration instead of Cisco ACI model internals.

Figure 19: Example of leaf node ports and FEX ports in Cisco ACI, on page 78 shows examples of Ethernet ports directly on leaf nodes or FEX modules attached to leaf nodes and how each is represented in the CLI. For FEX ports, the *fex-id* is included in the naming of the port itself as in **ethernet 101/1/1**. While describing an interface range, the **ethernet** keyword need not be repeated as in NX-OS. Example: **interface ethernet 101/1/1-2, 102/1/1-2**.

Figure 19: Example of leaf node ports and FEX ports in Cisco ACI



- Leaf node ID numbers are global.
- The *fex-id* numbers are local to each leaf node.
- Note the space after the keyword **ethernet**.

Procedure

Step 1 **configure**

Enters global configuration mode.

Example:

```
apic1# configure
```

Step 2 **leaf** *node-id*

Specifies the leaf nodes to be configured. The *node-id* can be a single node ID or a range of IDs, in the form *node-id1−node-id2*, to which the configuration will be applied.

Example:

```
apic1(config)# leaf 102
```

Step 3 **interface** *type*

Specifies the interface that you are configuring. You can specify the interface type and identity. For an Ethernet port, use "ethernet slot / port."

Example:

```
apic1(config-leaf)# interface ethernet 1/2
```

Step 4 (Optional) **fex associate** *node-id*

If the interface or interfaces to be configured are FEX interfaces, you must use this command to attach the FEX module to a leaf node before configuration.

Note
This step is required before creating a port channel using FEX ports.

Example:

```
apic1(config-leaf-if)# fex associate 101
```

Step 5 **speed** *speed*

The speed setting is shown as an example. At this point you can configure any of the interface settings shown in the table below.

Example:

```
apic1(config-leaf-if)# speed 10G
```

The following table shows the interface settings that can be configured at this point:

Command	Purpose
[no] shut	Shut down physical interface
[no] speed *speedValue*	Set the speed for physical interface
[no] link debounce time *time*	Set link debounce
[no] negotiate auto	Configure negotiate
[no] cdp enable	Disable/enable Cisco Discovery Protocol (CDP)
[no] mcp enable	Disable/enable Mis-cabling Protocol (MCP)

Command	Purpose
[no] lldp transmit	Set the transmit for physical interface
[no] lldp receive	Set the LLDP receive for physical interface
spanning-tree {bpduguard | bpdufilter} {enable | disable}	Configure spanning tree BPDU
[no] storm-control level *percentage* [burst-rate *percentage*]	Storm-control configuration (percentage)
[no] storm-control pps *packets-per-second* burst-rate *packets-per-second*	Storm-control configuration (packets-per-second)

Examples

Configure one port in a leaf node. The following example shows how to configure the interface eth1/2 in leaf 101 for the following properties: speed, cdp, and admin state.

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# cdp enable
apic1(config-leaf-if)# no shut
```

Configure multiple ports in multiple leaf nodes. The following example shows the configuration of speed for interfaces eth1/1-10 for each of the leaf nodes 101-103.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface eth 1/1-10
apic1(config-leaf-if)# speed 10G
```

Attach a FEX to a leaf node. The following example shows how to attach a FEX module to a leaf node. Unlike in NX-OS, the leaf node port Eth1/5 is implicitly configured as fabric port and a FEX fabric port channel is created internally with the FEX uplink port(s). In Cisco ACI, the FEX fabric port channels use default configuration and no user configuration is allowed.



Note This step is required before creating a port channel using FEX ports, as described in the next example.

```
apic1(config)# leaf 102
apic1(config-leaf)# interface eth 1/5
apic1(config-leaf-if)# fex associate 101
```

Configure FEX ports attached to leaf nodes. This example shows configuration of speed for interfaces eth1/1-10 in FEX module 101 attached to each of the leaf nodes 102-103. The FEX ID 101 is included in the port identifier. FEX IDs start with 101 and are local to a leaf node.

```
apic1(config)# leaf 102-103
apic1(config-leaf)# interface eth 101/1/1-10
apic1(config-leaf-if)# speed 1G
```

Port Channels

PC Host Load Balancing Algorithms

The following table provides the default hash algorithm and symmetric hash algorithm options used in port channel load balancing across Cisco Application Centric Infrastructure (ACI) leaf node downlinks. The symmetric hash algorithm options were introduced in Cisco Application Policy Infrastructure Controller (APIC) release 2.3(1e).

Table 5: PC Host Load Balancing Algorithms

Traffic Type	Hashing Data Points
End Host PC (default)	**For Layer 2 traffic**: • Source MAC address • Destination MAC address • Segment ID (VXLAN VNID) or VLAN ID **For IP Traffic**: • Source MAC address • Destination MAC address • Source IP address • Destination IP address • Protocol type • Source Layer 4 port • Destination Layer 4 port • Segment ID (VXLAN VNID) or VLAN ID
PC symmetric hash (configurable)	**Choose one option**: • Source IP address • Destination IP address • Source Layer 4 port • Destination Layer 4 port

When there is more than one port channel on a leaf switch, such as Po1 and Po2, then the following scenario is supported:

- Po1: Enable symmetric hash with SIP only.
- Po2: Do not enable symmetric hash. Use default hashing.

However, the following scenario is not supported because the second port channel Po2 has a different hash parameter:

- Po1: Enable symmetric hash with SIP only.
- Po2: Enable symmetric hash with DIP only.

That is, on a single leaf switch, all port channels that require symmetric hashing should use the same hash policy/parameter or use the default hashing.

Symmetric hashing is not supported on the following switches:

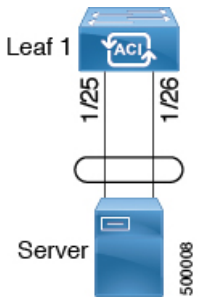
- Cisco Nexus 93128TX
- Cisco Nexus 9372PX
- Cisco Nexus 9372PX-E
- Cisco Nexus 9372TX
- Cisco Nexus 9372TX-E
- Cisco Nexus 9396PX
- Cisco Nexus 9396TX

Note Port channel hash algorithms are applied at each individual leaf node independently. The algorithms do not have influence on load balancing within the fabric, such as load balancing to leaf nodes in a vPC pair. Thus, symmetrical hashing is not supported on a vPC.

Cisco ACI Leaf Switch Port Channel Configuration Using the GUI

This procedure uses either the **Fabric** > **Access Policies** > **Quick Start** > **Configure Interfaces** or the **Fabric** > **Access Policies** > **Interface Configuration** page to attach a server to a Cisco Application Centric Infrastructure (ACI) leaf switch interface with a port channel. The steps would be the same for attaching other kinds of devices to a Cisco ACI leaf switch interface.

Figure 20: Switch Port Channel Configuration



Before you begin

- The Cisco ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.

- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the Cisco ACI fabric and available.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 Use one of the following methods to open the **Configure Interfaces** dialog:

Method 1:

a) In the **Navigation** pane, choose **Quick Start**.
b) In the **Work** pane, click **Configure Interfaces**.

Method 2:

a) In the **Navigation** pane, choose **Interface Configuration**.
b) In the **Work** pane, choose **Actions** > **Configure Interfaces**.

Step 3 In the **Configure Interfaces** dialog, perform the following actions:

a) For **Node Type**, click **Leaf**.
b) For **Port Type**, click **Access**.
c) For **Interface Type**, choose the desired type.
d) For **Interface Aggregation Type**, choose **PC**.
e) For **Node**, click **Select Node**, put a check in the box for the desired switch (node), then click **OK**.
f) For **Interfaces For All Switches**, enter the range of desired interfaces.
g) For **PC/vPC Interface Policy Group**, click **Select PC/vPC Interface Policy Group**, then select an existing port channel policy group or create new one.
h) For **Port Channel Member Policy**, click **Select Port Channel Member Policy**, then select an existing port channel member policy or create new one.
i) In the **Select PC/vPC Interface Policy Group** dialog, choose an existing policy group or click **Create PC/vPC Interface Policy Group** to create a new one.
j) Click **Save**.

What to do next

This completes the port channel configuration steps.



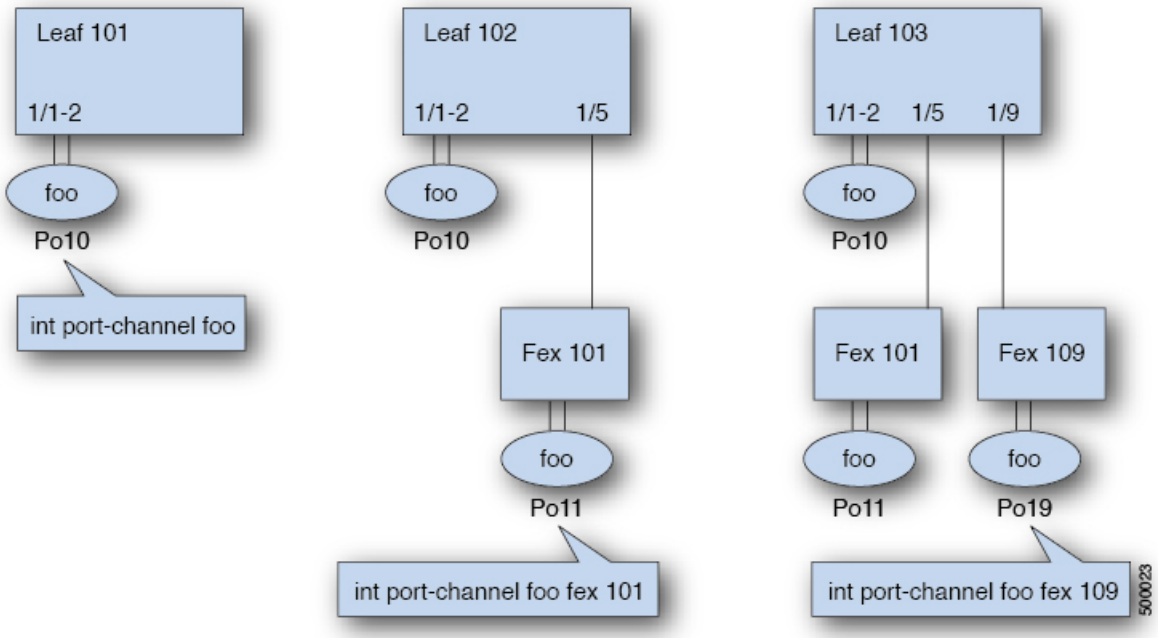
Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Port Channels in Leaf Nodes and FEX Devices Using the NX-OS CLI

Port channels are logical interfaces in NX-OS used to aggregate bandwidth for multiple physical ports and also for providing redundancy in case of link failures. In NX-OS, port channel interfaces are identified by user-specified numbers in the range 1 to 4096 unique within a node. Port channel interfaces are either configured explicitly (using the **interface port-channel** command) or created implicitly (using the **channel-group** command). The configuration of the port channel interface is applied to all the member ports of the port channel. There are certain compatibility parameters (speed, for example) that cannot be configured on the member ports.

In the ACI model, port channels are configured as logical entities identified by a name to represent a collection of policies that can be assigned to set of ports in one or more leaf nodes. Such assignment creates one port channel interface in each of the leaf nodes identified by an auto-generated number in the range 1 to 4096 within the leaf node, which may be same or different among the nodes for the same port channel name. The membership of these port channels may be same or different as well. When a port channel is created on the FEX ports, the same port channel name can be used to create one port channel interface in each of the FEX devices attached to the leaf node. Thus, it is possible to create up to N+1 unique port channel interfaces (identified by the auto-generated port channel numbers) for each leaf node attached to N FEX modules. This is illustrated with the examples below. Port channels on the FEX ports are identified by specifying the *fex-id* along with the port channel name (**interface port-channel foo fex 101**, for example).

Figure 21: Example with port channels on leaf switches and FEX ports



- N+1 instances per leaf of port channel foo are possible when each leaf is connected to N FEX nodes.
- Leaf ports and FEX ports cannot be part of the same port channel instance.
- Each FEX node can have only one instance of port channel foo.

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**template port-channel** *channel-name* **Example:** `apic1(config)# template port-channel foo`	Creates a new port channel or configures an existing port channel (global configuration).
Step 3	[**no**] **switchport access vlan** *vlan-id* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **Example:** `apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg webEpg`	Deploys the EPG with the VLAN on all ports with which the port channel is associated.

Step 4	**channel-mode active**
	Example:
	`apic1(config-po-ch-if)# channel-mode active`
	Note
	To enable symmetric hashing, enter the **lacp symmetric-hash** command:
	`apic1(config-po-ch-if)# lacp symmetric-hash`
		Note
		The **channel-mode** command is equivalent to the mode option in the channel-group command in NX-OS. In ACI, however, this is supported for the port channel (not on a member port).
		Symmetric hashing is not supported on the following switches:
		• Cisco Nexus 93128TX
		• Cisco Nexus 9372PX
		• Cisco Nexus 9372PX-E
		• Cisco Nexus 9372TX
		• Cisco Nexus 9372TX-E
		• Cisco Nexus 9396PX
		• Cisco Nexus 9396TX

Step 5	**exit** **Example:** `apic1(config-po-ch-if)# exit`	Returns to configure mode.
Step 6	**leaf** *node-id* **Example:** `apic1(config)# leaf 101`	Specifies the leaf switches to be configured. The *node-id* can be a single node ID or a range of IDs, in the form *node-id1−node-id2*, to which the configuration will be applied.

	Command or Action	Purpose
Step 7	**interface** *type* **Example:** `apic1(config-leaf)# interface ethernet 1/1-2`	Specifies the interface or range of interfaces that you are configuring to the port channel.
Step 8	[**no**] **channel-group** *channel-name* **Example:** `apic1(config-leaf-if)# channel-group foo`	Assigns the interface or range of interfaces to the port channel. Use the keyword **no** to remove the interface from the port channel. To change the port channel assignment on an interface, you can enter the **channel-group** command without first removing the interface from the previous port channel.

Step 9	(Optional) **lacp port-priority** *priority*
	Example:
	`apic1(config-leaf-if)# lacp port-priority 1000`
	`apic1(config-leaf-if)# lacp rate fast`
		This setting and other per-port LACP properties can be applied to member ports of a port channel at this point.
		Note
		In the ACI model, these commands are allowed only after the ports are member of a port channel. If a port is removed from a port channel, configuration of these per-port properties are removed as well.

The following table shows various commands for global configurations of port channel properties in the ACI model. These commands can also be used for configuring overrides for port channels in a specific leaf in the (config-leaf-if) CLI mode. The configuration made on the port channel is applied to all member ports.

CLI Syntax	Feature
[no] speed <speedValue>	Set the speed for port channel
[no] link debounce time <time>	Set Link Debounce for port channel
[no] negotiate auto	Configure Negotiate for port channel
[no] cdp enable	Disable/Enable CDP for port channel
[no] mcp enable	Disable/Enable MCP for port channel
[no] lldp transmit	Set the transmit for port channel
[no] lldp receive	Set the lldp receive for port channel
spanning-tree <bpduguard | bpdufilter> <enable | disable>	Configure spanning tree BPDU
[no] storm-control level <percentage> [burst-rate <percentage>]	Storm-control configuration (percentage)
[no] storm-control pps <packet-per-second> burst-rate <packets-per-second>	Storm-control configuration (packets-per-second)
[no] channel-mode { active | passive | on| mac-pinning }	LACP mode for the link in port channel l

CLI Syntax	Feature
[no] lacp min-links <value>	Set minimum number of links
[no] lacp max-links <value>	Set maximum number of links
[no] lacp fast-select-hot-standby	LACP fast select for hot standby ports
[no] lacp graceful-convergence	LACP graceful convergence
[no] lacp load-defer	LACP load defer member ports
[no] lacp suspend-individual	LACP individual Port suspension
[no] lacp port-priority	LACP port priority
[no] lacp rate	LACP rate

Examples

Configure a port channel (global configuration). A logical entity foo is created that represents a collection of policies with two configurations: speed and channel mode. More properties can be configured as required.

Note The channel mode command is equivalent to the mode option in the channel group command in NX-OS. In ACI, however, this supported for the port channel (not on member port).

```
apic1(config)# template port-channel foo
apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg
webEpg
apic1(config-po-ch-if)# speed 10G
apic1(config-po-ch-if)# channel-mode active
```

Configure ports to a port channel in a FEX. In this example, port channel foo is assigned to ports Ethernet 1/1-2 in FEX 101 attached to leaf node 102 to create an instance of port channel foo. The leaf node will auto-generate a number, say 1002 to identify the port channel in the switch. This port channel number would be unique to the leaf node 102 regardless of how many instance of port channel foo are created.

Note The configuration to attach the FEX module to the leaf node must be done before creating port channels using FEX ports.

```
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group foo
```

In Leaf 102, this port channel interface can be referred to as interface port channel foo FEX 101.

```
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel foo fex 101
apic1(config-leaf)# shut
```

Configure ports to a port channel in multiple leaf nodes. In this example, port channel foo is assigned to ports Ethernet 1/1-2 in each of the leaf nodes 101-103. The leaf nodes will auto generate a number unique in each node (which may be same or different among nodes) to represent the port channel interfaces.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo
```

Add members to port channels. This example would add two members eth1/3-4 to the port channel in each leaf node, so that port channel foo in each node would have members eth 1/1-4.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
```

Remove members from port channels. This example would remove two members eth1/2, eth1/4 from the port channel foo in each leaf node, so that port channel foo in each node would have members eth 1/1, eth1/3.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface eth 1/2,1/4
apic1(config-leaf-if)# no channel-group foo
```

Configure port channel with different members in multiple leaf nodes. This example shows how to use the same port channel foo policies to create a port channel interface in multiple leaf nodes with different member ports in each leaf. The port channel numbers in the leaf nodes may be same or different for the same port channel foo. In the CLI, however, the configuration will be referred as interface port channel foo. If the port channel is configured for the FEX ports, it would be referred to as interface port channel foo fex <fex-id>.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 103
apic1(config-leaf)# interface ethernet 1/5-8
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group foo
```

Configure per port properties for LACP. This example shows how to configure member ports of a port channel for per-port properties for LACP.

Note In ACI model, these commands are allowed only after the ports are member of a port channel. If a port is removed from a port channel, configuration of these per-port properties would be removed as well.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo
```

```
apic1(config-leaf-if)# lacp port-priority 1000
apic1(config-leaf-if)# lacp rate fast
```

Configure admin state for port channels. In this example, a port channel foo is configured in each of the leaf nodes 101-103 using the channel-group command. The admin state of port channel(s) can be configured in each leaf using the port channel interface. In ACI model, the admin state of the port channel cannot be configured in the global scope.

```
// create port-channel foo in each leaf
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo

// configure admin state in specific leaf
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel foo
apic1(config-leaf-if)# shut
```

Override config is very helpful to assign specific vlan-domain, for example, to the port channel interfaces in each leaf while sharing other properties.

```
// configure a port channel global config
apic1(config)# interface port-channel foo
apic1(config-if)# speed 1G
apic1(config-if)# channel-mode active

// create port-channel foo in each leaf
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/1-2
apic1(config-leaf-if)# channel-group foo

// override port-channel foo in leaf 102
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel foo
apic1(config-leaf-if)#  speed 10G
apic1(config-leaf-if)#  channel-mode on
apic1(config-leaf-if)#  vlan-domain dom-foo
```

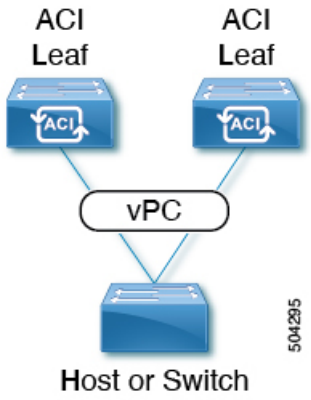
This example shows how to change port channel assignment for ports using the channel-group command. There is no need to remove port channel membership before assigning to other port channel.

```
apic1(config)# leaf 101-103
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group foo
apic1(config-leaf-if)# channel-group bar
```

Virtual Port Channels in Cisco ACI

A virtual port channel (vPC) allows links that are physically connected to two different Cisco Application Centric Infrastructure (ACI) leaf nodes to appear as a single port channel (PC) to a third device, such as a network switch, server, any other networking device that supports link aggregation technology. vPCs consist of two Cisco ACI leaf switches designated as vPC peer switches. Of the vPC peers, one is primary and one is secondary. The system formed by the switches is referred to as a vPC domain.

Figure 22: vPC Domain



The following behavior is specific to the Cisco ACI vPC implementation:

- No dedicated peer-link between the vPC peers. Instead, the fabric itself serves as the Multi-Chassis Trunking (MCT).
- Peer reachability protocol: Cisco ACI uses the Zero Message Queue (ZMQ) instead of Cisco Fabric Services (CFS).
  - ZMQ is an open-source, high-performance messaging library that uses TCP as the transport.
  - This library is packaged as libzmq on the switch and linked into each application that needs to communicate with a vPC peer.
- Peer reachability is not handled using a physical peer link. Instead, routing triggers are used to detect peer reachability.
  - The vPC manager registers with Unicast Routing Information Base (URIB) for peer route notifications.
  - When IS-IS discovers a route to the peer, URIB notifies the vPC manager, which in turn attempts to open a ZMQ socket with the peer.
  - When the peer route is withdrawn by IS-IS, the vPC manager is again notified by URIB, and the vPC manager brings down the MCT link.
- When creating a vPC domain between two leaf switches, the following hardware model limitations apply:
  - Generation 1 switches are compatible only with other generation 1 switches. These switch models can be identified by the lack of "EX," "FX, "FX2," "GX," or later suffix at the end of the switch name. For example, N9K-9312TX.
  - Generation 2 and later switches can be mixed together in a vPC domain. These switch models can be identified by the "EX," "FX, "FX2," "GX," or later suffix at the end of the switch name. For example N9K-93108TC-EX or N9K-9348GC-FXP.

  Examples of compatible vPC switch pairs:
  - N9K-C9312TX and N9K-C9312TX
  - N9K-C93108TC-EX and N9K-C9348GC-FXP

- N9K-C93180TC-FX and N9K-C93180YC-FX
- N9K-C93180YC-FX and N9K-C93180YC-FX

Examples of incompatible vPC switch pairs:

- N9K-C9312TX and N9K-C93108TC-EX
- N9K-C9312TX and N9K-C93180YC-FX

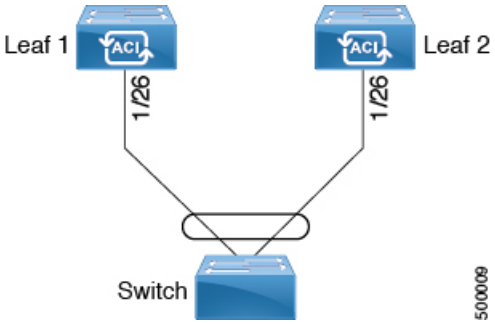
- Port channels and virtual port channels can be configured with or without LACP.

  If you configure a virtual port channel with LACP, LACP sets a port to the suspended state if it does not receive an LACP PDU from the peer. This can cause some servers to fail to boot up as they require LACP to bring up the port logically. You can tune the behavior to individual use by disabling **LACP suspend individual**. To do so, create a port channel policy in your vPC policy group, and after setting the mode to LACP active, remove **Suspend Individual Port**. Afterward, the ports in the vPC will stay active and continue to send LACP packets.

- Adaptive load balancing (ALB), based on ARP negotiation, across virtual port channels is not supported in Cisco ACI.

Cisco ACI Virtual Port Channel Workflow

Figure 23: Virtual port channel configuration



The configuration workflow for virtual port channels (vPCs) is as follows:

Before you begin

- Ensure that you have read/write access privileges to the infra security domain.
- Ensure that the target leaf switches with the necessary interfaces are available.
- Ensure that the hardware of the two leaf switches that are going to be part of the same vPC pair is compatible. For more information, see Virtual Port Channels in Cisco ACI, on page 89.

Procedure

Step 1 Configure the VLAN pools, domain, AAEP, access leaf port policy group of type vPC.

Step 2 Configure the vPC switch pairs.

Step 3 Configure the vPC interfaces.

Step 4 Configure the application profile.

a) On the menu bar, choose **Tenants** > **All Tenants**.

b) In the Work pane, double-click a tenant.

c) In the Navigation pane, choose ***tenant_name*** > **Quick Start**.

d) Configure the endpoint groups (EPGs), contracts, bridge domain, subnet, and context.

e) Associate the application profile EPGs with the virtual port channel switch profile that you created previously.

Defining a vPC Using the GUI

This procedure defines a vPC using the GUI. We recommend that you keep the leaf switch peer group names simple as shown in the following example:

- Leaf201_202
- Leaf203_204
- Leaf205_206

For naming and numbering best practices, see the *Cisco ACI Object Naming and Numbering: Best Practices* document:

https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/kb/b-Cisco-ACI-Naming-and-Numbering.html

Before you begin

Ensure that the hardware of the two leaf switches that are going to be part of the same vPC pair is compatible. For more information, see Virtual Port Channels in Cisco ACI, on page 89.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the Navigation pane, choose **Policies** > **Switch** > **Virtual Port Channel default**.

Step 3 In the **Explicit vPC Protection Groups** table, click **+** and fill out the fields as follows:

a) In the **Name** field, enter the vPC pair name.

Example name: `Leaf201_202`. A name similar to the example easily identifies which two fabric nodes are vPC peers.

b) In the **ID** field, enter the vPC pair ID (logical peer ID).

Example ID: `201`. The example uses the first node ID number of the pair to make it easier to correlate the ID with the vPC pair.

c) In the **Switch 1** and **Switch 2** fields, choose the leaf switches for the vPC switch pair.

d) Click **Submit**.

The vPC pair gets added to the **Explicit vPC Protection Groups** table. The **Virtual IP** value is an auto-generated IP address from the system tunnel endpoint (TEP) pool, and represents the virtual shared (Anycast) TEP of the vPC switch pair. That is, packets destined to vPC-connected endpoints of the vPC pair will use this Anycast VTEP to send the packets.

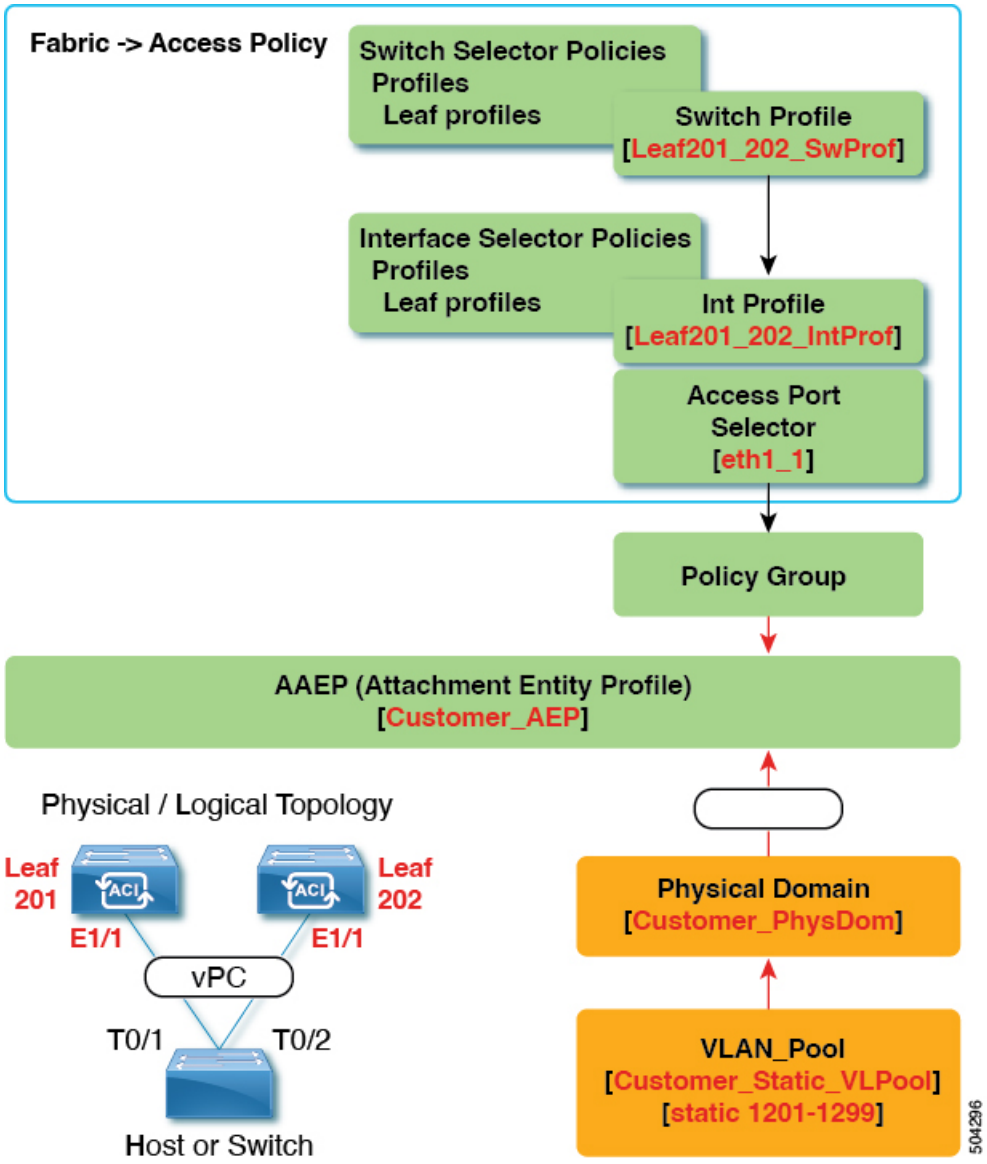
Configuring Virtual Port Channels in Leaf Nodes and FEX Devices Using Profiles and Selectors

vPC With the Same Leaf Switch Interfaces Across Two Leaf Switches With Combined Profiles

For the example of this use case, you define the following things:

- A combined switch profile called `Leaf201_202_SwProf` (node 201 and node 202).
- A combined interface profile called `Leaf201_202_IntProf` (node 201 and node 202).
- An access port selector called `Eth1_1` (under the `Leaf201_202` interface profile) is pointing toward a vPC interface policy group.
- The vPC interface policy group is pointing toward an AAEP called `Customer_AEP`.
- The AEP (`Customer_AEP`) has an association with the `Customer_PhysDom`.
- The `Customer_PhysDom` has an association with a VLAN pool called `Customer_Static_VLPool`.

Figure 24: vPC With the Same Leaf Switch Interfaces Across Two Leaf Switches With Combined Profiles



What This Configuration Does

On switches `Leaf201` and `Leaf202`, configure port Eth1/1 to be part of a vPC. This vPC interface will have access to VLANs 1201 through 1299. Depending on the interface policy group, you can enable LACP Active and other interface specific policy configurations.

When to Use This Configuration

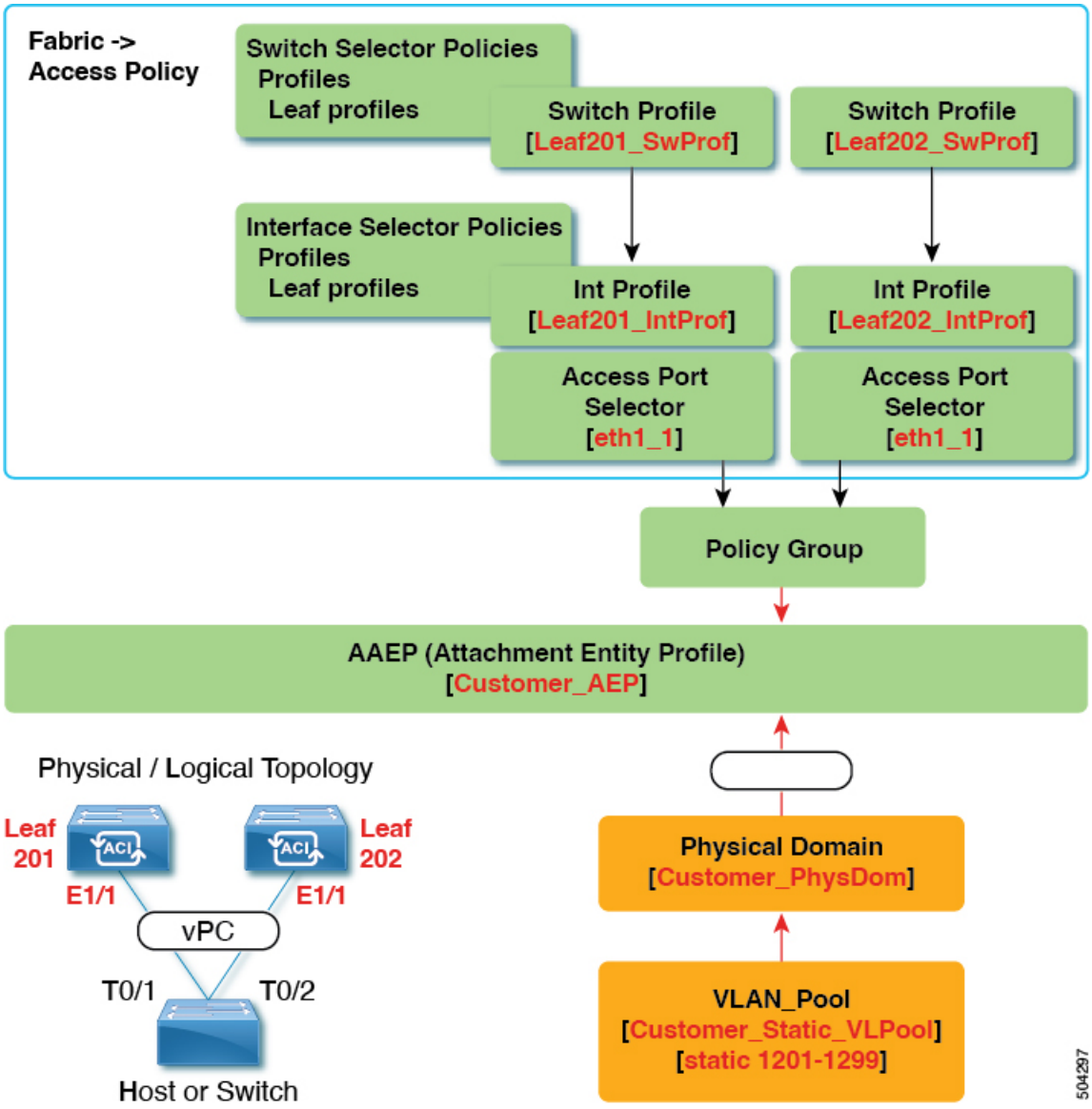
If you have dedicated pairs of compute leaf switches with nothing but vPC-connected servers, for example, this would be a solid use case for using combined-switch/interface profiles under your fabric access policies for those switches. You could preconfigure your switch, interface, access port selector, and vPC interface policy groups in such a way that allowed you to plug in 48 chassis-type servers with minimal effort.

vPC With the Same Leaf Switch Interfaces Across Two Leaf Switches with Individual Profiles

For the example of this use case, you define the following things:

- Individual switch profiles called `Leaf201_SwProf` and `Leaf202_SwProf` (node 201 and node 202).
- Individual interface profiles called `Leaf201_IntProf` and `Leaf202_IntProf` (node 201 and node 202)
- Access port selectors called `Eth1_1` (under the `Leaf201` and `Leaf202` interface profiles) is pointing toward the same vPC interface policy group.
- The vPC interface policy group is pointing toward an AAEP called `Customer_AEP`.
- The AEP (`Customer_AEP`) has an association with the `Customer_PhysDom`.
- The `Customer_PhysDom` has an association with a VLAN pool called `Customer_Static_VLPool`.

Figure 25: vPC With the Same Leaf Switch Interfaces Across Two Leaf Switches with Individual Profiles



What This Configuration Does

On switches `Leaf201` and `Leaf202`, configure port Eth1/1 to be a part of a vPC. This vPC interface will have access to VLANs 1201 through 1299. Depending on the interface policy group, you can enable LACP active and other interface specific policy configurations.

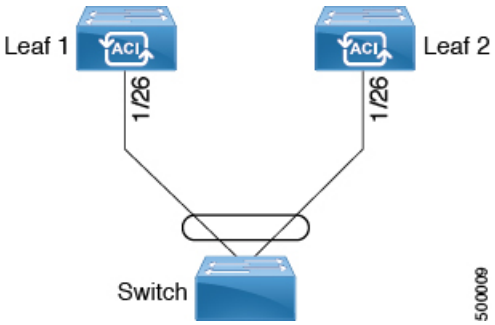
When to Use This Configuration

Use this configuration when you have leaf switches that support mixed workloads, such as compute, services, or Cisco Application Policy Infrastructure Controllers (APICs). In this case, having individual interface profiles allows for the most amount of flexibility, while allowing you to keep your **Fabric** > **Access Policies** configuration as clean and manageable as possible.

Configuring a Cisco ACI Leaf Switch Virtual Port Channel Using the Interface Configuration Model Using the GUI

This procedure uses the "Interface Configuration" methodology to attach a trunked switch to a Cisco Application Centric Infrastructure (ACI) leaf switch virtual port channel. The steps are the same for attaching other kinds of devices to an Cisco ACI leaf switch interface.

Figure 26: Switch Virtual Port Channel Configuration



Before you begin

- The Cisco ACI fabric is installed, the Cisco Application Policy Infrastructure Controllers (APICs) are online, and the Cisco APIC cluster is formed and healthy.
- A Cisco APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the Cisco ACI fabric and available.

Note When creating a vPC domain between two leaf switches, ensure that the hardware of the two leaf switches that are going to be part of the same vPC pair is compatible. For more information, see Virtual Port Channels in Cisco ACI, on page 89.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 Use one of the following methods to open the **Configure Interfaces** dialog:

Method 1:

a) In the **Navigation** pane, choose **Quick Start**.
b) In the **Work** pane, click **Configure Interfaces**.

Method 2:

a) In the **Navigation** pane, choose **Interface Configuration**.
b) In the **Work** pane, choose **Actions** > **Configure Interfaces**.

Step 3 In the **Configure Interfaces** dialog, perform the following actions:

a) For **Node Type**, click **Leaf**.
b) For **Port Type**, click **Access**.
c) For **Interface Type**, click **Ethernet**.
d) For **Interface Aggregation Type**, choose **vPC**.
e) For **vPC Leaf Switch Pair**, click **Select vPC Leaf Switch Pair**, put a check in the box for the desired switch pair, then click **Select**. You can select multiple switches. Optionally, click **Create vPC Leaf Switch Pair** and fill out the fields as desired, then select the pair and click **Select**.
f) For **Interfaces For All Switches**, enter the range of desired interfaces.
g) For **PC/vPC Interface Policy Group**, click **Select PC/vPC Interface Policy Group**.
h) In the **Select PC/vPC Interface Policy Group** dialog, choose an existing vPC policy group and click **Select**. Optionally, click **Create PC/vPC Interface Policy Group** to create a new vPC policy group, fill out the fields, and click **Save**, then choose that policy group and click **Select**.
i) For **Port Channel Member Policy**, click **Select Port Channel Member Policy**, then choose a policy and click **Select**. Optionally, click **Create Port Channel Member Policy**, fill out the fields as desired, and click **Save**, then choose that policy and click **Select**..
j) Click **Save**.

Verification: Use the CLI **show int** command on the leaf switches where the external switch is attached to verify that the vPC is configured accordingly.

What to do next

This completes the switch virtual port channel configuration steps.



Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring Virtual Port Channels in Leaf Nodes and FEX Devices Using the NX-OS CLI

A virtual port channel (vPC) is an enhancement to port-channels that allows connection of a host or switch to two upstream leaf nodes to improve bandwidth utilization and availability. In NX-OS, vPC configuration is done in each of the two upstream switches and configuration is synchronized using peer link between the switches.

The Cisco Application Centric Infrastructure (ACI) model does not require a peer link and vPC configuration can be done globally for both the upstream leaf nodes. A global configuration mode called **vpc context** is introduced in Cisco ACI and vPC interfaces are represented using a type **interface vpc** that allows global configuration applicable to both leaf nodes.

Two different topologies are supported for vPC in the Cisco ACI model: vPC using leaf ports and vPC over FEX ports. It is possible to create many vPC interfaces between a pair of leaf nodes and similarly, many vPC interfaces can be created between a pair of FEX modules attached to the leaf node pairs in a straight-through topology.

vPC considerations include:

- The vPC name used is unique between leaf node pairs. For example, only one vPC 'corp' can be created per leaf pair (with or without FEX).
- Leaf ports and FEX ports cannot be part of the same vPC.
- Each FEX module can be part of only one instance of vPC corp.
- vPC context allows configuration
- The vPC context mode allows configuration of all vPCs for a given leaf pair. For vPC over FEX, the *fex-id* pairs must be specified either for the vPC context or along with the vPC interface, as shown in the following two alternative examples.

```
(config)# vpc context leaf 101 102
(config-vpc)# interface vpc Reg fex 101 101
```

or

```
(config)# vpc context leaf 101 102 fex 101 101
(config-vpc)# interface vpc Reg
```

In the Cisco ACI model, vPC configuration is done in the following steps (as shown in the examples below).

Note A VLAN domain is required with a VLAN range. It must be associated with the port-channel template.

1. VLAN domain configuration (global config) with VLAN range
2. vPC domain configuration (global config)
3. Port-channel template configuration (global config)
4. Associate the port channel template with the VLAN domain
5. Port-channel configuration for vPC (global config)
6. Configure ports to vPC in leaf nodes
7. Configure Layer 2, Layer 3 for vPC in the vPC context

Before you begin

Ensure that the hardware of the two leaf switches that are going to be part of the same vPC pair is compatible. For more information, see Virtual Port Channels in Cisco ACI, on page 89.

Procedure

Step 1 **configure**

Enters global configuration mode.

Example:

```
apic1# configure
```

Step 2 **vlan-domain***name*[**dynamic**] [**type** *domain-type*]

Configures a VLAN domain for the virtual port-channel (here with a port-channel template).

Example:

```
apic1(config)# vlan-domain dom1 dynamic
```

Step 3 **vlan***range*

Configures a VLAN range for the VLAN domain and exits the configuration mode. The range can be a single VLAN or a range of VLANs.

Example:

```
apic1(config-vlan)# vlan 1000-1999
apic1(config-vlan)# exit
```

Step 4 **vpc domain explicit** *domain-id* **leaf** *node-id1 node-id2*

Configures a vPC domain between a pair of leaf nodes. You can specify the vPC domain ID in the explicit mode along with the leaf node pairs.

Alternative commands to configure a vPC domain are as follows:

- **vpc domain** [**consecutive** | **reciprocal**]

  The consecutive and reciprocal options allow auto configuration of a vPC domain across all leaf nodes in the Cisco ACI fabric.

- **vpc domain consecutive** *domain-start* **leaf** *start-node end-node*

  This command configures a vPC domain consecutively for a selected set of leaf node pairs.

Example:

```
apic1(config)# vpc domain explicit 1 leaf 101 102
```

Step 5 **peer-dead-interval** *interval*

Configures the time delay the Leaf switch waits to restore the vPC before receiving a response from the peer. If it does not receive a response from the peer within this time, the Leaf switch considers the peer dead and brings up the vPC with the role as a master. If it does receive a response from the peer it restores the vPC at that point. The range is from 5 seconds to 600 seconds. The default is 200 seconds.

Example:

```
apic1(config-vpc)# peer-dead-interval 10
```

Step 6 **exit**

Returns to global configuration mode.

Example:

```
apic1(config-vpc)# exit
```

Step 7 **template port-channel** *channel-name*

Creates a new port-channel or configures an existing port-channel (global configuration).

All vPCs are configured as port-channels in each leaf pair. The same port-channel name must be used in a leaf pair for the same vPC. This port-channel can be used to create a vPC among one or more pairs of leaf nodes. Each leaf node will have only one instance of this vPC.

Example:

```
apic1(config)# template port-channel corp
```

Step 8 **vlan-domain member***vlan-domain-name*

Associates the port channel template with the previously configured VLAN domain.

Example:

```
vlan-domain member dom1
```

Step 9 **switchport access vlan** *vlan-id* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

Deploys the EPG with the VLAN on all ports with which the port-channel is associated.

Example:

```
apic1(config-po-ch-if)# switchport access vlan 4 tenant ExampleCorp application Web epg webEpg
```

Step 10 **channel-mode active**

Note
A port-channel must be in active channel-mode for a vPC.

Example:

```
apic1(config-po-ch-if)# channel-mode active
```

Step 11 **exit**

Returns to configure mode.

Example:

```
apic1(config-po-ch-if)# exit
```

Step 12 **leaf** *node-id1 node-id2*

Specifies the pair of leaf switches to be configured.

Example:

```
apic1(config)# leaf 101-102
```

Step 13 **interface** *typeleaf/interface-range*

Specifies the interface or range of interfaces that you are configuring to the port-channel.

Example:

```
apic1(config-leaf)# interface ethernet 1/3-4
```

Step 14 [**no**] **channel-group** *channel-name* **vpc**

Assigns the interface or range of interfaces to the port-channel. Use the keyword **no** to remove the interface from the port-channel. To change the port-channel assignment on an interface, you can enter the **channel-group** command without first removing the interface from the previous port-channel.

Note
The **vpc** keyword in this command makes the port-channel a vPC. If the vPC does not already exist, a vPC ID is automatically generated and is applied to all member leaf nodes.

Example:

```
apic1(config-leaf-if)# channel-group corp vpc
```

Step 15 **exit**

Example:

```
apic1(config-leaf-if)# exit
```

Step 16 **exit**

Example:

```
apic1(config-leaf)# exit
```

Step 17 **vpc context leaf** *node-id1 node-id2*

The vPC context mode allows configuration of vPC to be applied to both leaf node pairs.

Example:

```
apic1(config)# vpc context leaf 101 102
```

Step 18 **interface vpc** *channel-name*

Example:

```
apic1(config-vpc)# interface vpc blue fex 102 102
```

Step 19 (Optional) [**no**] **shutdown**

Administrative state configuration in the vPC context allows changing the admin state of a vPC with one command for both leaf nodes.

Example:

```
apic1(config-vpc-if)# no shut
```

Example

This example shows how to configure a basic vPC.

```
apic1# configure
apic1(config)# vlan-domain dom1 dynamic
apic1(config-vlan)# vlan 1000-1999
apic1(config-vlan)# exit
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config-vpc)# peer-dead-interval 10
apic1(config-vpc)# exit
apic1(config)# template port-channel corp
apic1(config-po-ch-if)# vlan-domain member dom1
apic1(config-po-ch-if)# channel-mode active
apic1(config-po-ch-if)# exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group corp vpc
apic1(config-leaf-if)# exit
apic1(config)# vpc context leaf 101 102
```

This example shows how to configure vPCs with FEX ports.

```
apic1(config-leaf)# interface ethernet 101/1/1-2
apic1(config-leaf-if)# channel-group Reg vpc
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc corp
apic1(config-vpc-if)# exit
```

```
apic1(config-vpc)# interface vpc red fex 101 101
apic1(config-vpc-if)# switchport
apic1(config-vpc-if)# exit
apic1(config-vpc)# interface vpc blue fex 102 102
apic1(config-vpc-if)# shut
```

Reflective Relay (802.1Qbg)

Reflective relay is a switching option beginning with Cisco APIC Release 2.3(1). Reflective relay—the tagless approach of IEEE standard 802.1Qbg—forwards all traffic to an external switch, which then applies policy and sends the traffic back to the destination or target VM on the server as needed. There is no local switching. For broadcast or multicast traffic, reflective relay provides packet replication to each VM locally on the server.

One benefit of reflective relay is that it leverages the external switch for switching features and management capabilities, freeing server resources to support the VMs. Reflective relay also allows policies that you configure on the Cisco APIC to apply to traffic between the VMs on the same server.

In the Cisco ACI, you can enable reflective relay, which allows traffic to turn back out of the same port it came in on. You can enable reflective relay on individual ports, port channels, or virtual port channels as a Layer 2 interface policy using the APIC GUI, NX-OS CLI, or REST API. It is disabled by default.

The term *Virtual Ethernet Port Aggregator* (VEPA) is also used to describe 802.1Qbg functionality.

Reflective Relay Support

Reflective relay supports the following:

- IEEE standard 802.1Qbg tagless approach, known as reflective relay.

  Cisco APIC Release 2.3(1) release does not support the IEE standard 802.1Qbg S-tagged approach with multichannel technology.

- Physical domains.

  Virtual domains are not supported.

- Physical ports, port channels (PCs), and virtual port channels (vPCs).

  Cisco Fabric Extender (FEX) and blade servers are not supported. If reflective relay is enabled on an unsupported interface, a fault is raised, and the last valid configuration is retained. Disabling reflective relay on the port clears the fault.

- Cisco Nexus 9000 series switches with *EX* or *FX* at the end of their model name.

Enabling Reflective Relay Using the GUI

Reflective relay is disabled by default; however, you can enable it on a port, port channel, or virtual port channel as a Layer 2 interface policy on the switch. You first configure a policy and then associate the policy with a policy group.

Before you begin

This procedure assumes that you have set up the Cisco Application Centric Infrastructure (ACI) fabric and installed the physical switches.

Procedure

Step 1 Choose **Fabric** > **External Access Policies** > > **Interface Policies** and then open the **Policies** folder.
Step 2 Right-click the **L2 Interface** folder and choose **Create L2 Interface Policy**.
Step 3 In the **Create L2 Interface Policy** dialog box, enter a name in the **Name** field.
Step 4 In the **Reflective Relay (802.1Qbg)** area, click **enabled**.
Step 5 Choose other options in the dialog box as needed.
Step 6 Click **SUBMIT**.
Step 7 In the **Policies** navigation pane, open the **Policy Groups** folder and click the **Leaf Policy Groups** folder.
Step 8 In the **Leaf Policy Groups** central pane, expand the **ACTIONS** drop-down list, and choose **Create Leaf Access Port Policy Group**, **Create PC Interface Policy Group**, **Create vPC Interface Policy Group**, or **Create PC/vPC Override Policy Group**.
Step 9 In the policy group dialog box, enter a name in the **Name field**.
Step 10 From the **L2 Interface Policy** drop-down list, choose the policy that you just created to enable Reflective Relay.
Step 11 Click **Submit**.

Enabling Reflective Relay Using the NX-OS CLI

Reflective relay is disabled by default; however, you can enable it on a port, port channel, or virtual port channel as a Layer 2 interface policy on the switch. In the NX-OS CLI, you can use a template to enable reflective relay on multiple ports or you can enable it on individual ports.

Before you begin

This procedure assumes that you have set up the Cisco Application Centric Infrastructure (ACI) fabric and installed the physical switches.

Procedure

Enable reflective relay on one or multiple ports:

Example:

This example enables reflective relay on a single port:

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# switchport vepa enabled
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

Example:

This example enables reflective relay on multiple ports using a template:

```
apic1(config)# template policy-group grp1
apic1(config-pol-grp-if)# switchport vepa enabled
apic1(config-pol-grp-if)# exit
```

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/2-4
apic1(config-leaf-if)# policy-group grp1
```

Example:

This example enables reflective relay on a port channel:

```
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel po2
apic1(config-leaf-if)# switchport vepa enabled
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)#
```

Example:

This example enables reflective relay on multiple port channels:

```
apic1(config)# template port-channel po1
apic1(config-if)# switchport vepa enabled
apic1(config-if)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/3-4
apic1(config-leaf-if)# channel-group po1
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

Example:

This example enables reflective relay on a virtual port channel:

```
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config-vpc)# exit
apic1(config)# template port-channel po4
apic1(config-if)# exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface eth 1/11-12
apic1(config-leaf-if)# channel-group po4 vpc
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc po4
apic1(config-vpc-if)# switchport vepa enabled
```

Configuring Port, PC, and vPC Connections to FEX Devices

FEX connections and the profiles used to configure them can be created using the GUI, NX-OS-style CLI, or the REST API. Interface profiles for configuring FEX connections are supported since the Cisco Application Policy Infrastructure Controller (APIC) 3.0(1k) release.

The configuration consists of two key steps:

- Defining the connectivity between the FEX and the Cisco Application Centric Infrastructure (ACI) leaf switch
- Configuring the FEX ports connected to the servers

After you configure the FEX connectivity to the Cisco ACI leaf switch, the configuration of the FEX host-facing ports is no different than the configuration of Cisco ACI leaf switch ports as individual interfaces, port channels, or vPCs.

For information on how to configure ports, PCs, and vPCs using the GUI, the NX-OS-style CLI, or the REST API, see the following sections:

- Physical Ports Configuration, on page 74
- Port Channels, on page 81
- Virtual Port Channels in Cisco ACI, on page 89

ACI FEX Guidelines

Observe the following guidelines when deploying a FEX:

- Assuming that no leaf switch front panel ports are configured to deploy and EPG and VLANs, a maximum of 10,000 port EPGs are supported for being deployed using a FEX.
- For each FEX port or vPC that includes FEX ports as members, a maximum of 20 EPGs per VLAN are supported.
- A vPC with FEX interfaces ignores the minimum and maximum number of links configured in its port-channel policy. The vPC remains up even if the number of links is less than the minimum or greater than the maximum.
- Configuring port-channel across two Fexes on the same leaf switch is not supported.

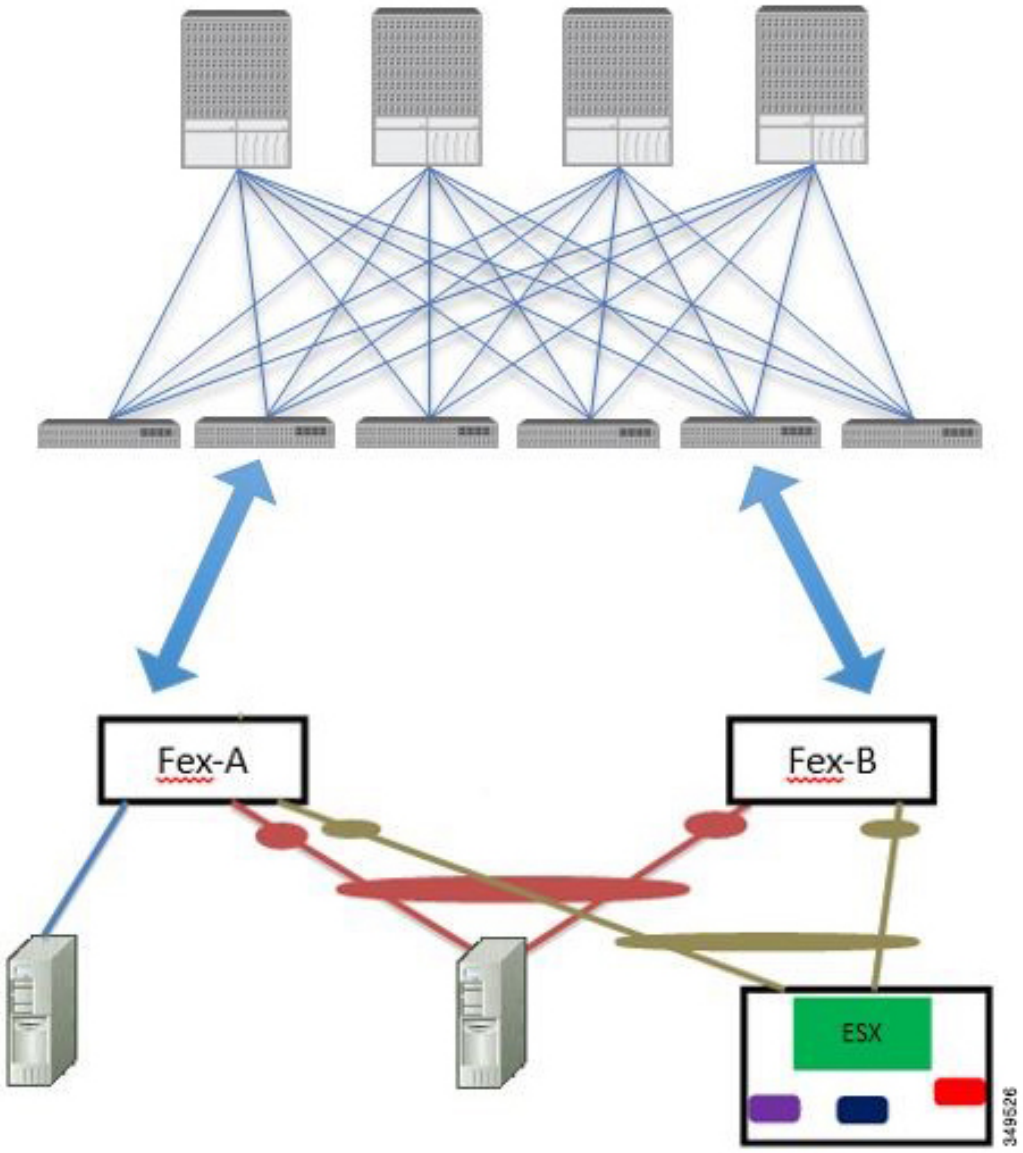
FEX Virtual Port Channels

The ACI fabric supports Cisco Fabric Extender (FEX) server-side virtual port channels (vPC), also known as an FEX straight-through vPC.



Note When creating a vPC domain between two leaf switches, ensure that the hardware of the two leaf switches that are going to be part of the same vPC pair is compatible. For more information, see Virtual Port Channels in Cisco ACI, on page 89.

Figure 27: Supported FEX vPC Topologies



Supported FEX vPC port channel topologies include the following:

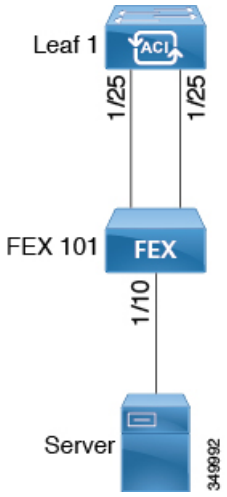
- Both VTEP and non-VTEP hypervisors behind a FEX.
- Virtual switches (such as AVS or VDS) connected to two FEXs that are connected to the ACI fabric (vPCs directly connected on physical FEX ports is not supported - a vPC is supported only on port channels).

Note When using GARP as the protocol for notification of IP-to-MAC binding changes to different interfaces on the same FEX, you must set the bridge domain mode to **ARP Flooding** and enable **EP Move Detection Mode**: **GARP-based Detection**, on the **L3 Configuration** page of the bridge domain wizard. This workaround is only required with Generation 1 switches. With Generation 2 switches or later, this is not an issue.

Configuring a FEX Connection to the ACI Leaf Switch Using the GUI

This procedure provides the steps for attaching a server to the FEX. The steps would be the same for attaching any device to a Cisco Application Centric Infrastructure (ACI)-attached FEX.

Figure 28: Basic FEX Configuration



Note Configuring FEX connections with FEX IDs `165` to `199` is not supported in the APIC GUI. To use one of these FEX IDs, configure the profile using the NX-OS style CLI. For more information, see *Configuring FEX Connections Using Interface Profiles with the NX-OS Style CLI*.

Before you begin

- The Cisco ACI fabric is installed, Cisco Application Policy Infrastructure Controllers (APICs) are online, and the Cisco APIC cluster is formed and healthy.
- An Cisco APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches, interfaces, and protocols are configured and available.
- The FEX is powered on and connected to the target leaf switch interfaces

Note A maximum of eight members are supported in fabric port channels connected to FEXs.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the **Navigation** pane, choose **Interface Configuration**.

Step 3 In the **Work** pane, choose **Actions** > **Fabric Extender**.

Step 4 In the **Fabric Extender** dialog, perform the following actions:

a) For **Node**, click **Select Node**, put a check in the box for the desired node, then click **OK**. You can select multiple nodes.

b) For **Interfaces For All Switches**, enter the range of desired interfaces.

c) For **Connected FEX ID**, enter the ID of the FEX.

You must configure FEX IDs 165 - 199, using the NX-OS-style CLI. See *Configuring FEX Connections Using Interface Profiles with the NX-OS Style CLI*.

d) Click **Save**.
The Cisco APIC auto-generates the necessary FEX profile (`switch-policy-name_FexPFEX-ID`) and selector (`switch-policy-name_ifselctor`).

Verification: Use the CLI **show fex** command on the switch where the FEX is attached to verify that the FEX is online.

Step 5 You can now configure FEX interfaces, such as regular Cisco ACI leaf switch interfaces by using **Fabric** > **Fabric Access** > **Interface Configuration**.

What to do next

Note While this configuration enables hardware connectivity, no data traffic can flow without a valid application profile, EPG, and contract that is associated with this hardware configuration.

Configuring FEX Connectivity to an ACI leaf switch Using Profiles with the NX-OS-Style CLI

Use this procedure to configure FEX connections to leaf nodes using the NX-OS style CLI.

Note Configuring FEX connections with FEX IDs `165` to `199` is not supported in the Cisco Application Policy Infrastructure Controller (APIC) GUI. To use one of these FEX IDs, configure the profile using the following commands.

Procedure

Step 1 **configure**

Enters global configuration mode.

Example:

```
apic1# configure
```

Step 2 **leaf-interface-profile** *name*

Specifies the leaf interface profile to be configured.

Example:

```
apic1(config)# leaf-interface-profile fexIntProf1
```

Step 3 **leaf-interface-group** *name*

Specifies the interface group to be configured.

Example:

```
apic1(config-leaf-if-profile)# leaf-interface-group leafIntGrp1
```

Step 4 **fex associate** *fex-id* [**template** *template-typefex-template-name*]

Attaches a FEX module to a leaf node. Use the optional template keyword to specify a template to be used. If it does not exist, the system creates a template with the name and type you specified.

Example:

```
apic1(config-leaf-if-group)# fex associate 101
```

Example

This merged example configures a leaf interface profile for FEX connections with ID 101.

```
apic1# configure
apic1(config)# leaf-interface-profile fexIntProf1
apic1(config-leaf-if-profile)# leaf-interface-group leafIntGrp1
apic1(config-leaf-if-group)# fex associate 101
```

Configuring Port Profiles

Uplink and downlink conversion is supported on Cisco Nexus 9000 series switches with names that end in EX or FX, and later (for example, N9K-C9348GC-FXP or N9K-C93240YC-FX2). A FEX connected to converted downlinks is also supported.

For information about the supported supported Cisco switches, see Port Profile Configuration Summary, on page 115.

When an uplink port is converted to a downlink port, it acquires the same capabilities as any other downlink port.

Restrictions

- Fast Link Failover policies and port profiles are not supported on the same port. If port profile is enabled, Fast Link Failover cannot be enabled or vice versa.
- The last 2 uplink ports of supported leaf switches cannot be converted to downlink ports (they are reserved for uplink connections).
- Dynamic breakouts (both 100Gb and 40Gb) are supported on profiled QSFP ports on the N9K-C93180YC-FX switch. Breakout and port profile are supported together for conversion of uplink to downlink on ports 49-52. Breakout (both **10g-4x** and **25g-4x** options) is supported on downlink profiled ports.
- The N9K-C9348GC-FXP does not support FEX.
- Breakout is supported only on downlink ports, and not on fabric ports that are connected to other switches.
- A Cisco ACI leaf switch cannot have more than 56 fabric links.
- Reloading a switch after changing a switch's port profile configuration interrupts traffic through the data plane.
- If a port profile is configured on any LEM type and you want to replace that LEM, the replacement LEM type must match the LEM type you removed.

Guidelines

In converting uplinks to downlinks and downlinks to uplinks, consider these guidelines.

Subject	Guideline
Port profile guidelines for N9K-X9400-8D	The 6.0(2) release adds suport for this LEM and these guidelines apply: • This LEM has 8 ports with a default of 4 downlinks and 4 uplinks. • Port profile conversion is supported on the first 6 ports. • This LEM does not have a port group dependency.
Port profile guidelines for N9K-X9400-16W	The 6.0(2) release adds suport for this LEM and these guidelines apply: • This LEM has 16 ports, with a default of 12 downlinks and 4 uplinks. • Port profile conversion is supported on the first 6 ports. • This LEM has a port group dependency of 2 ports for port profile conversion. Which means that ports 1-2, 3-4, and 5-6 have a port group dependency. For example, if port 2 is to be converted as an uplink, port 1 should also be converted to an uplink.

Subject	Guideline
Decommissioning nodes with port profiles	If a decommissioned node has the Port Profile feature deployed on it, the port conversions are not removed even after decommissioning the node. It is necessary to manually delete the configurations after decommission, for the ports to return to the default state. To do this, log onto the switch, run the `setup-clean-config.sh` script, and wait for it to run. Then, enter the `reload` command. Optionally, you can specify `-k` with the `setup-clean-config.sh` script to allow the port-profile setting to persist across the reload, making an additional reboot unnecessary. Beginning with 6.0(5), the port-profile setting persists across the reload when running the `setup-clean-config.sh` script with no option, `-k` or `--keep-port-profile`. To manually delete the configuration, run the `setup-clean-config.sh` script with `-d` or `--delete-profiles`.
Maximum uplink port limit	When the maximum uplink port limit is reached and ports 25 and 27 are converted from uplink to downlink and back to uplink on Cisco 93180LC-EX switches: On Cisco N9K-93180LC-EX switches, ports 25 and 27 are the original uplink ports. Using the port profile, if you convert port 25 and 27 to downlink ports, ports 29, 30, 31, and 32 are still available as four original uplink ports. Because of the threshold on the number of ports (which is maximum of 12 ports) that can be converted, you can convert 8 more downlink ports to uplink ports. For example, ports 1, 3, 5, 7, 9, 13, 15, 17 are converted to uplink ports and ports 29, 30, 31 and 32 are the 4 original uplink ports (the maximum uplink port limit on Cisco 93180LC-EX switches). When the switch is in this state and if the port profile configuration is deleted on ports 25 and 27, ports 25 and 27 are converted back to uplink ports, but there are already 12 uplink ports on the switch (as mentioned earlier). To accommodate ports 25 and 27 as uplink ports, 2 random ports from the port range 1, 3, 5, 7, 9, 13, 15, 17 are denied the uplink conversion and this situation cannot be controlled by the user. Therefore, it is mandatory to clear all the faults before reloading the leaf node to avoid any unexpected behavior regarding the port type. It should be noted that if a node is reloaded without clearing the port profile faults, especially when there is a fault related to limit-exceed, the port might not be in an expected operational state.

Breakout Limitations

Switch	Releases	Limitations
N9K-C93180LC-EX	Cisco APIC 3.1(1) and later	• 40Gb and 100Gb dynamic breakouts are supported on ports 1 through 24 on odd numbered ports. • When the top ports (odd ports) are broken out, then the bottom ports (even ports) are error disabled. • Port profiles and breakouts are not supported on the same port. However, you can apply a port profile to convert a fabric port to a downlink, and then apply a breakout configuration.
N9K-C9336C-FX2-E	Cisco APIC 5.2(4) and later	• 40Gb and 100Gb dynamic breakouts are supported on ports 1 through 34. • A port profile cannot be applied to a port with breakout enabled. However, you can apply a port profile to convert a fabric port to a downlink, and then apply a breakout configuration. • All 34 ports can be configured as breakout ports. • If you want to apply a breakout configuration on 34 ports, you must configure a port profile on the ports to have 34 downlink ports, then you must reboot the leaf switch. • If you apply a breakout configuration to a leaf switch for multiple ports at the same time, it can take up to 10 minutes for the hardware of 34 ports to be programmed. The ports remain down until the programming completes. The delay can occur for a new configuration, after a clean reboot, or during switch discovery.

Switch	Releases	Limitations
N9K-C9336C-FX2	Cisco APIC 4.2(4) and later	• 40Gb and 100Gb dynamic breakouts are supported on ports 1 through 34. • A port profile cannot be applied to a port with breakout enabled. However, you can apply a port profile to convert a fabric port to a downlink, and then apply a breakout configuration. • All 34 ports can be configured as breakout ports. • If you want to apply a breakout configuration on 34 ports, you must configure a port profile on the ports to have 34 downlink ports, then you must reboot the leaf switch. • If you apply a breakout configuration to a leaf switch for multiple ports at the same time, it can take up to 10 minutes for the hardware of 34 ports to be programmed. The ports remain down until the programming completes. The delay can occur for a new configuration, after a clean reboot, or during switch discovery.
N9K-C9336C-FX2	Cisco APIC 3.2(1) up through, but not including, 4.2(4)	• 40Gb and 100Gb dynamic breakouts are supported on ports 1 through 30. • Port profiles and breakouts are not supported on the same port. However, you can apply a port profile to convert a fabric port to a downlink, and then apply a breakout configuration. • A maximum of 20 ports can be configured as breakout ports.

Switch	Releases	Limitations
N9K-C93180YC-FX	Cisco APIC 3.2(1) and later	• 40Gb and 100Gb dynamic breakouts are supported on ports 49 though 52, when they are on profiled QSFP ports. To use them for dynamic breakout, perform these steps: • Convert ports 49-52 to front panel ports (downlinks). • Perform a port-profile reload, using one of these methods: • In the Cisco APIC GUI, navigate to **Fabric** > **Inventory** > **Pod** > **Leaf**, right-click **Chassis** and choose **Reload**. • In the iBash CLI, enter the **reload** command. • Apply breakouts on the profiled ports 49-52. • Ports 53 and 54 do not support either port profiles or breakouts.
N9K-C93240YC-FX2	Cisco APIC 4.0(1) and later	Breakout is not supported on converted downlinks.

Port Profile Configuration Summary

The following table summarizes supported uplinks and downlinks for the switches that support port profile conversions from uplink to downlink and downlink to uplink.

Switch Model	Default Links	Max Uplinks (Fabric Ports)	Max Downlinks (Server Ports)	Release Supported
N9K-C9348GC-FXP[1] N9K-C9348GC-FX3	48 x 100M/1G BASE-T downlinks 4 x 10/25 Gbps SFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	48 x 100M/1G BASE-T downlinks 4 x 10/25 Gbps SFP28 uplinks 2 x 40/100 Gbps QSFP28 uplinks	Same as default port configuration	3.1(1) 6.0(5)

Switch Model	Default Links	Max Uplinks (Fabric Ports)	Max Downlinks (Server Ports)	Release Supported
N9K-C93180LC-EX	24 x 40 Gbps QSFP28 downlinks (ports 1-24) 2 x 40/100 Gbps QSFP28 uplinks (ports 25, 27) 4 x 40/100 Gbps QSFP28 uplinks (ports 29-32) Or 12 x 100 Gbps QSFP28 downlinks (odd number ports from 1-24) 2 x 40/100 Gbps QSFP28 uplinks (ports 25, 27) 4 x 40/100 Gbps QSFP28 uplinks (ports 29-32)	18 x 40 Gbps QSFP28 downlinks (from 1-24) 6 x 40 Gbps QSFP28 uplinks(from 1-24) 2 x 40/100 Gbps QSFP28 uplinks(25, 27) 4 x 40/100 Gbps QSFP28 uplinks(29-32) Or 6 x 100 Gbps QSFP28 downlinks(odd number from 1-24) 6 x 100 Gbps QSFP28 uplinks(odd number from 1-24) 2 x 40/100 Gbps QSFP28 uplinks(25, 27) 4 x 40/100 Gbps QSFP28 uplinks(29-32)	24 x 40 Gbps QSFP28 downlinks(1-24) 2 x 40/100 Gbps QSFP28 downlinks(25, 27) 4 x 40/100 Gbps QSFP28 uplinks(29-32) Or 12 x 100 Gbps QSFP28 downlinks(odd number from 1-24) 2 x 40/100 Gbps QSFP28 downlinks (25, 27) 4 x 40/100 Gbps QSFP28 uplinks(29-32)	3.1(1)
N9K-C93180YC-EX N9K-C93180YC-FX	48 x 10/25 Gbps fiber downlinks 6 x 40/100 Gbps QSFP28 uplinks	Same as default port configuration	48 x 10/25 Gbps fiber downlinks 4 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	3.1(1)
		48 x 10/25 Gbps fiber uplinks 6 x 40/100 Gbps QSFP28 uplinks		4.0(1)
N9K-C93180YC-FX3	48 x 10/25 Gbps fiber downlinks 6 x 40/100 Gbps QSFP28 uplinks	18 x 10/25 Gbps fiber downlinks 30 x 10/25 Gbps fiber uplinks 6 x 40/100 Gbps QSFP28 uplinks	48 x 10/25 Gbps fiber downlinks 4 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	5.1(3)

Switch Model	Default Links	Max Uplinks (Fabric Ports)	Max Downlinks (Server Ports)	Release Supported
N9K-C93108TC-EX[2] N9K-C93108TC-FX[2] N9K-C93108TC-FX3	48 x 10GBASE-T downlinks 6 x 40/100 Gbps QSFP28 uplinks	Same as default port configuration	48 x 10/25 Gbps fiber downlinks 4 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	3.1(1) 4.0(1) 5.1(3)
N9K-C9336C-FX2	30 x 40/100 Gbps QSFP28 downlinks 6 x 40/100 Gbps QSFP28 uplinks	18 x 40/100 Gbps QSFP28 downlinks 18 x 40/100 Gbps QSFP28 uplinks	Same as default port configuration	3.2(1)
		18 x 40/100 Gbps QSFP28 downlinks 18 x 40/100 Gbps QSFP28 uplinks	34 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	3.2(3)
		36 x 40/100 Gbps QSFP28 uplinks	34 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	4.1(1)
N9K-C9336C-FX2-E	30 x 40/100 Gbps QSFP28 downlinks 6 x 40/100 Gbps QSFP28 uplinks	36 x 40/100 Gbps QSFP28 uplinks	34 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	5.2(4)
N9K-93240YC-FX2	48 x 10/25 Gbps fiber downlinks 12 x 40/100 Gbps QSFP28 uplinks	Same as default port configuration	48 x 10/25 Gbps fiber downlinks 10 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	4.0(1)
		48 x 10/25 Gbps fiber uplinks 12 x 40/100 Gbps QSFP28 uplinks		4.1(1)
N9K-C93216TC-FX2	96 x 10G BASE-T downlinks 12 x 40/100 Gbps QSFP28 uplinks	Same as default port configuration	96 x 10G BASE-T downlinks 10 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	4.1(2)

Switch Model	Default Links	Max Uplinks (Fabric Ports)	Max Downlinks (Server Ports)	Release Supported
N9K-C93360YC-FX2	96 x 10/25 Gbps SFP28 downlinks 12 x 40/100 Gbps QSFP28 uplinks	44 x 10/25Gbps SFP28 downlinks 52 x 10/25Gbps SFP28 uplinks 12 x 40/100Gbps QSFP28 uplinks	96 x 10/25 Gbps SFP28 downlinks 10 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	4.1(2)
N9K-C93600CD-GX	28 x 40/100 Gbps QSFP28 downlinks (ports 1-28) 8 x 40/100/400 Gbps QSFP-DD uplinks (ports 29-36)	28 x 40/100 Gbps QSFP28 uplinks 8 x 40/100/400 Gbps QSFP-DD uplinks	28 x 40/100 Gbps QSFP28 downlinks 6 x 40/100/400 Gbps QSFP-DD downlinks 2 x 40/100/400 Gbps QSFP-DD uplinks	4.2(2)
N9K-C9364C-GX	48 x 40/100 Gbps QSFP28 downlinks (ports 1-48) 16 x 40/100 Gbps QSFP28 uplinks (ports 49-64)	56 x 40/100 Gbps QSFP28 uplinks	62 x 40/100 Gbps QSFP28 downlinks 2 x 40/100 Gbps QSFP28 uplinks	4.2(3)
N9K-C9316D-GX	12 x 40/100/400 Gbps QSFP-DD downlinks (ports 1-12) 4 x 40/100/400 Gbps QSFP-DD uplinks (ports 13-16)	16 x 40/100/400 Gbps QSFP-DD uplinks	14 x 40/100/400 Gbps QSFP-DD downlinks	5.1(4)
N9K-C9332D-GX2B	2 x 1/10 Gbps SFP+ downlinks (ports 33-34) 24 x 40/100/400 Gbps QSFP-DD downlinks (ports 1-24) 8 x 40/100/400 Gbps QSFP-DD uplinks (ports 25-32)	2 x 1/10 Gbps SFP+ downlinks 32 x 40/100/400 Gbps QSFP-DD uplinks	2 x 1/10 Gbps SFP+ downlinks 30 x 40/100/400 Gbps QSFP-DD downlinks 2 x 40/100/400 Gbps QSFP-DD uplinks	5.2(3)

Switch Model	Default Links	Max Uplinks (Fabric Ports)	Max Downlinks (Server Ports)	Release Supported
N9K-C9348D-GX2A	2 x 1/10 Gbps SFP+ downlinks (ports 49-50) 36 x 40/100/400 Gbps QSFP-DD downlinks (ports 1-36) 12 x 40/100/400 Gbps QSFP-DD uplinks (ports 37-48)	2 x 1/10 Gbps SFP+ downlinks 48 x 40/100/400 Gbps QSFP-DD uplinks	2 x 1/10 Gbps SFP+ downlinks 46 x 40/100/400 Gbps QSFP-DD downlinks 2 x 40/100/400 Gbps QSFP-DD uplinks	5.2(5)
N9K-C9364D-GX2A	2 x 1/10 Gbps SFP+ downlinks (ports 65-66) 48 x 40/100/400 Gbps QSFP-DD downlinks (ports 1-48) 16 x 40/100/400 Gbps QSFP-DD uplinks (ports 49-64)	2 x 1/10 Gbps SFP+ downlinks 56 x 40/100/400 Gbps QSFP-DD uplinks	2 x 1/10 Gbps SFP+ downlinks 62 x 40/100/400 Gbps QSFP-DD downlinks 2 x 40/100/400 Gbps QSFP-DD uplinks	5.2(5)
N9K-C9408 with N9K-X9400-8D[3]	6 x 40/100/400 Gbps QSFP-DD downlinks 2 x 40/100/400 Gbps QSFP-DD uplinks	8 x 40/100/400 Gbps QSFP-DD uplinks	Same as default port configuration	6.0(2)
N9K-C9408 with N9K-X9400-16W[3]	12 x 100/200 Gbps QSFP56 downlinks 4 x 100/200 Gbps QSFP56 uplinks	6 x 100/200 Gbps QSFP56 uplinks (ports 1-6) 6 x 100/200 Gbps QSFP56 downlinks (ports 7-12) 4 x 100/200 Gbps QSFP56 uplinks (ports 13-16)	Same as default port configuration	6.0(2)[4]

1 Does not support FEX.

2 Only uplink to downlink conversion is supported.

3 Only ports 1 through 6 support port profile conversion.

4 The 6.0(2) release does not support 200 Gbps.

Changing an Uplink to a Downlink or Downlink to an Uplink Using the GUI

This procedure explains how to configure a port profile, which determines the port type: uplink or downlink. You can configure ports as uplinks or downlinks by using **Fabric** > **Access Policies** > **Interface Configuration** > **Actions** > **Convert Interfaces**. You can also use **Fabric** > **Inventory** > **Topology** > **Convert Interfaces**. The two methods provide the same workflow.

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, Cisco Application Policy Infrastructure Controller (APIC) are online, and the Cisco APIC cluster is formed and healthy.
- An Cisco APIC fabric administrator account is available that will enable creating or modifying the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the Cisco ACI fabric and available.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the Navigation pane, choose **Interface Configuration**.

Step 3 In the Work pane, choose **Actions** > **Convert Interfaces**.

Step 4 In the **Interface Configuration Support Type** drop-down list, choose **Convert to Uplink** or **Convert to Downlink**.

Step 5 For the **Node** field, click **Select Node** and select the nodes

Step 6 In the **Interfaces for All Switches** field, enter the desired interfaces.

After converting a downlink to uplink or uplink to downlink, you must reload the switch using the GUI or CLI `reload` command. Power cycling the switch will not work.

Changing an Uplink to a Downlink or Downlink to an Uplink Using the NX-OS-Style CLI

To configure a port profile using the NX-OS-style CLI, perform the following steps:

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, Cisco Application Policy Infrastructure Controller (APIC) are online, and the Cisco APIC cluster is formed and healthy.
- An Cisco APIC fabric administrator account is available that will enable creating or modifying the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the Cisco ACI fabric and available.

Procedure

Step 1 **configure**

Enters global configuration mode.

Example:

```
apic1# configure
```

Step 2 **leaf** *node-id*

Specifies the leaf switches to be configured.

Example:

```
apic1(config)# leaf 102
```

Step 3 **interface** *type*

Specifies the interface that you are configuring. You can specify the interface type and identity. For an Ethernet port, use ethernet *slot* / *port*.

Example:

```
apic1(config-leaf)# interface ethernet 1/2
```

Step 4 **port-direction** {**uplink** | **downlink**}

Determines the port direction or changes it. This example configures the port to be a downlink.

On the N9K-C9336C-FX switch, changing a port from uplink to downlink is not supported.

Example:

```
apic1(config-leaf-if)# port-direction downlink
```

Step 5 Log on to the leaf switch where the port is located and enter the **reload** command.

Verifying Port Profile Configuration and Conversion Using the NX-OS Style CLI

You can verify the configuration and the conversion of the ports using the **show interface brief** CLI command.



Note Port profile can be deployed only on the top ports of a Cisco N9K-C93180LC-EX switch, for example, 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, and 23. When the top port is converted using the port profile, the bottom ports are hardware disabled. For example, if Eth 1/1 is converted using the port profile, Eth 1/2 is hardware disabled.

Procedure

Step 1 This example displays the output for converting an uplink port to downlink port. Before converting an uplink port to downlink port, the output is displayed in the example. The keyword **routed** denotes the port as uplink port.

Example:

```
switch# show interface brief
<snip>
Eth1/49          --      eth  routed  down   sfp-missing             100G(D)   --
Eth1/50          --      eth  routed  down   sfp-missing             100G(D)   --
<snip>
```

Step 2 After configuring the port profile and reloading the switch, the output is displayed in the example. The keyword **trunk** denotes the port as downlink port.

Example:

```
switch# show interface brief
<snip>
Eth1/49          0       eth  trunk   down   sfp-missing             100G(D)   --
Eth1/50          0       eth  trunk   down   sfp-missing             100G(D)   --
<snip>
```

Editing an Interface Configuration

This procedure describes how to edit the configuration of an interface that you previously configured, which enables you to change the port policy group or description of the interface.

Before you begin

You must have at least one interface configured.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

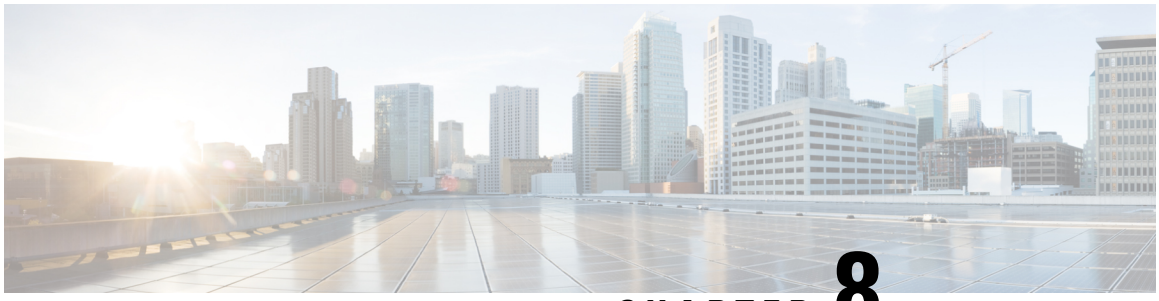
Step 2 In the Navigation pane, choose **Interface Configuration**.

Step 3 In the work pane, click the **...** at the right end of the row for the interface whose configuration you want to edit, then choose **Edit Interface Configuration**.

Step 4 In the **Edit Policy Group for *interface-name*** dialog, change the configuration as required.

Step 5 Click **Save**.

Note
For any existing configurations done using the node or port profile, you can migrate the entire FEX configuration using the APIC REST API Configuration Procedures.



CHAPTER 8

FCoE Connections

This chapter contains the following sections:

- Supporting Fibre Channel over Ethernet Traffic on the Cisco ACI Fabric , on page 123
- Fibre Channel over Ethernet Guidelines and Limitations, on page 125
- Fibre Channel over Ethernet Supported Hardware, on page 125
- Configuring FCoE Using the APIC GUI, on page 126
- Configuring FCoE Using the NX_OS Style CLI, on page 142
- SAN Boot with vPC, on page 152

Supporting Fibre Channel over Ethernet Traffic on the Cisco ACI Fabric

Cisco Application Centric Infrastructure (ACI) enables you to configure and manage support for Fibre Channel over Ethernet (FCoE) traffic on the Cisco ACI fabric.

FCoE is a protocol that encapsulates Fibre Channel packets within Ethernet packets, thus enabling storage traffic to move seamlessly between a Fibre Channel SAN and an Ethernet network.

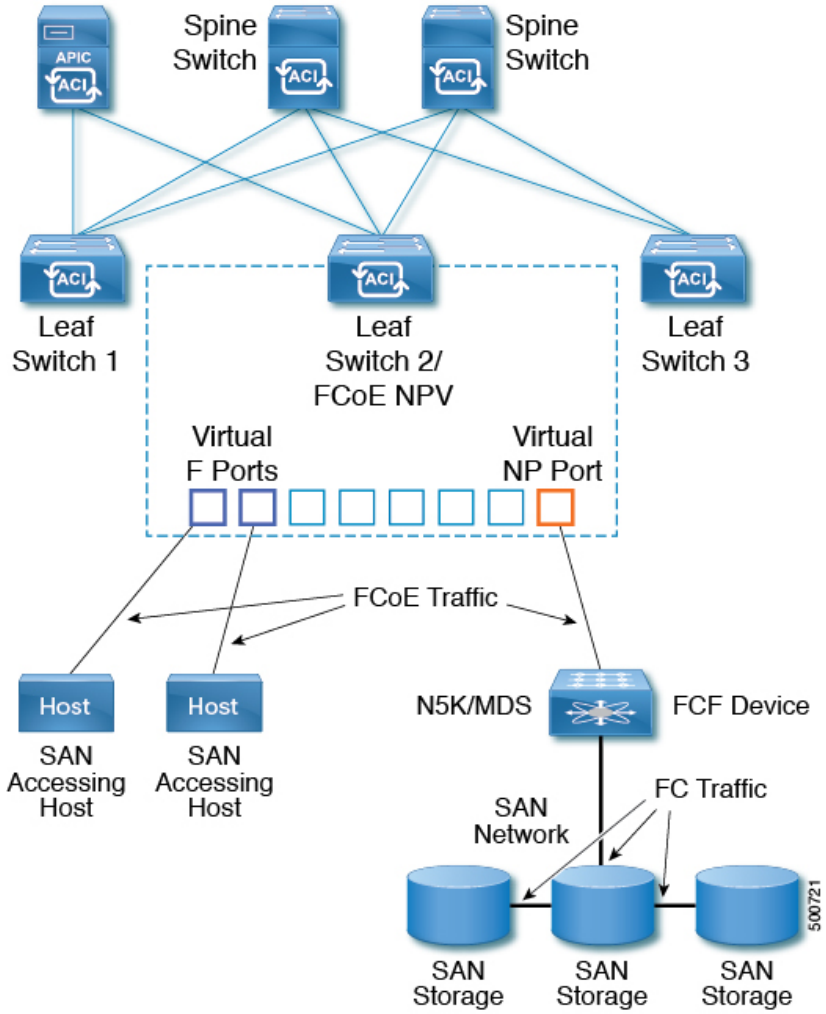
A typical implementation of FCoE protocol support on the Cisco ACI fabric enables hosts located on the Ethernet-based Cisco ACI fabric to communicate with SAN storage devices located on a Fibre Channel network. The hosts are connecting through virtual F ports deployed on an Cisco ACI leaf switch. The SAN storage devices and Fibre Channel network are connected through a Fibre Channel Forwarding (FCF) bridge to the Cisco ACI fabric through a virtual NP port, deployed on the same Cisco ACI leaf switch as is the virtual F port. Virtual NP ports and virtual F ports are also referred to generically as virtual Fibre Channel (vFC) ports.

Note In the FCoE topology, the role of the Cisco ACI leaf switch is to provide a path for FCoE traffic between the locally connected SAN hosts and a locally connected FCF device. The leaf switch does not perform local switching between SAN hosts, and the FCoE traffic is not forwarded to a spine switch.

Topology Supporting FCoE Traffic Through Cisco ACI

The topology of a typical configuration supporting FCoE traffic over the Cisco ACI fabric consists of the following components:

Figure 29: Cisco ACI Topology Supporting FCoE Traffic



- One or more Cisco ACI leaf switches configured through Fibre Channel SAN policies to function as an NPV backbone.
- Selected interfaces on the NPV-configured leaf switches configured to function as virtual F ports, which accommodate FCoE traffic to and from hosts running SAN management or SAN-consuming applications.
- Selected interfaces on the NPV-configured leaf switches configured to function as virtual NP ports, which accommodate FCoE traffic to and from a Fibre Channel Forwarding (FCF) bridge.

The FCF bridge receives Fibre Channel traffic from Fibre Channel links typically connecting SAN storage devices and encapsulates the Fibre Channel packets into FCoE frames for transmission over the Cisco ACI fabric to the SAN management or SAN Data-consuming hosts. It receives FCoE traffic and repackages it back to the Fibre Channel for transmission over the Fibre Channel network.

Note In the above Cisco ACI topology, FCoE traffic support requires direct connections between the hosts and virtual F ports and direct connections between the FCF device and the virtual NP port.

Cisco Application Policy Infrastructure Controller (APIC) servers enable an operator to configure and monitor the FCoE traffic through the Cisco APIC GUI, or NX-OS-style CLI, or through application calls to the REST API.

Topology Supporting FCoE Initialization

In order for FCoE traffic flow to take place as described, you must also set up separate VLAN connectivity over which SAN Hosts broadcast FCoE Initialization protocol (FIP) packets to discover the interfaces enabled as F ports.

vFC Interface Configuration Rules

Whether you set up the vFC network and EPG deployment through the Cisco APIC GUI, NX-OS-style CLI, or the REST API, the following general rules apply across platforms:

- F port mode is the default mode for vFC ports. NP port mode must be specifically configured in the Interface policies.
- The load balancing default mode is for leaf-switch or interface level vFC configuration is src-dst-ox-id.
- One VSAN assignment per bridge domain is supported.
- The allocation mode for VSAN pools and VLAN pools must always be static.
- vFC ports require association with a VSAN domain (also called Fibre Channel domain) that contains VSANs mapped to VLANs.

Fibre Channel over Ethernet Guidelines and Limitations

The VLAN used for FCoE should have vlanScope set to `Global`. Setting vlanScope to `portLocal` is not supported for FCoE. The value is set using the Layer 2 interface policy (l2IfPol).

Fibre Channel over Ethernet Supported Hardware

FCoE is supported on the following switches:

- N9K-C93180LC-EX

  When 40 Gigabit Ethernet (GE) ports are enabled as FCoE F or NP ports, they cannot be enabled for 40GE port breakout. FCoE is not supported on breakout ports.
- N9K-C93108TC-FX
- N9K-C93108TC-EX (only FCoE NPV)
- N9K-C93180YC-EX
- N9K-C93180LC-EX

  Support includes FCoE on FEX ports.
- N9K-C93180YC-FX

Support includes 10/25G ports (1-48), 40G ports (1/49-54), 4x10G breakout ports (1/49-54), and FCoE on FEX ports.

FCoE is supported on the following Nexus FEX devices:

- N2K-C2348UPQ-10GE
- N2K-C2348TQ-10GE
- N2K-C2232PP-10GE
- N2K-B22DELL-P
- N2K-B22HP-P
- N2K-B22IBM-P
- N2K-B22DELL-P-FI

Configuring FCoE Using the APIC GUI

FCoE GUI Configuration

FCoE Policy, Profile, and Domain Configurations

You can use the APIC GUI under the Fabric Access Policies tab to configure policies, policy groups, and profiles to enable customized and scaled-out deployment and assignment of FCoE supporting F and NP ports on your ACI leaf switches. Then, under the APIC the Tenant tab, you can configure EPG access to those ports.

Policies and Policy Groups

APIC policies and policy groups you create or configure for FCoE support include the following:

Access Switch Policy Group

The combination of switch-level policies that support FCoE traffic through ACI leaf switches.

You can associate this policy group with a leaf profile to enable FCoE support on designated ACI leaf switches.

This policy group consists of the following policies:

- **Fibre Channel SAN Policy**

  Specifies the EDTOV, RATOV, and MAC Address prefix (also called the FC map) values used by the NPV leaf.

- **Fibre Channel Node Policy**

  Specifies the load balance options and FIP keep alive intervals that apply to FCoE traffic associated with this switch policy group.

Interface Policy Groups

The combination of interface-level policies that support FCoE traffic through interfaces on ACI leaf switches.

You can associate this policy group with an FCoE supportive interface profile to enable FCoE support on designated interfaces.

You configure two interface policy groups: One policy group for F ports, and one policy group for NP ports.

The following policies in the interface policy group apply to FCoE enablement and traffic:

- **Priority Flow Control Policy**

  Specifies the state of priority flow control (PFC) on the interfaces to which this policy group is applied.

  This policy specifies under what circumstances QoS-level priority flow control will be applied to FCoE traffic.

- **Fibre Channel Interface Policy**

  Specifies whether the interfaces to which this policy group is applied are to be configured as F ports or NP ports.

- **Slow Drain Policy**

  Specifies the policy for handling FCoE packets that are causing traffic congestion on the ACI Fabric.

Global Policies

The APIC global policies whose settings can affect the performance characteristics of FCoE traffic on the ACI fabric.

The Global **QOS Class Policies** for **Level1**, **Level2**, **Level4**, **Level5**, or **Level6** connections, contain the following settings that affect FCoE traffic on the ACI fabric:

- **PFC Admin State must be set to Auto**

  Specifies whether to enable priority flow control to this level of FCoE traffic (default value is false).

- **No Drop COS**

  Specifies whether to enable a no-drop policy for this level of FCoE traffic designated with a certain Class of Service (CoS) level.

  Note: QoS level enabled for PFC and FCoE no-drop must match with the Priority Group ID enabled for PFC on CNA.

  Note: Only one QoS level can be enabled for no-drop and PFC, and the same QoS level must be associated with FCoE EPGs.

- QoS Class—Priority flow control requires that CoS levels be globally enabled for the fabric and assigned to the profiles of applications that generate FCoE traffic.

  CoS Preservation must also be enabled—Navigate to **Fabric** > **Access Policies** > **Policies** > **Global** > **QoS Class** and enable **Preserve COS Dot1p Preserve**.

Note Some legacy CNAs may require the **Level2** Global QoS Policy to be used as the **No Drop** PFC, FCoE (Fibre Channel over Ethernet) QoS Policy. If your Converged Network Adapters (CNAs) are not logging into the fabric, and you have noticed that no FCoE Initiation Protocol (FIP) frames are being sent by the CNAs, try enabling **Level2** as the FCoE QoS policy. The **Level2** policy must be attached to the FCoE EPGs in use and only one QoS level can be enabled for PFC no-drop.

Profiles

APIC profiles that you can create or configure for FCoE support include the following:

Leaf Profile

Specifies the ACI Fabric leaf switches on which to configure support of FCoE traffic.

The combination of policies contained in the access switch policy group can be applied to the leaf switches included in this profile.

Interface Profiles

Specifies a set of interfaces on which to deploy F Ports or NP Ports.

You configure at least two leaf interface profiles: One interface profile for F ports, and one interface profile for NP ports.

The combination of policies contained in the interface policy group for F ports can be applied to the set of interfaces included in the interface profile for F ports.

The combination of policies contained in the interface policy group for NP ports can be applied to the set of interfaces included in the interface profile for NP ports.

Attached Entity Profile

Binds the interface policy group settings with the Fibre Channel domain mapping.

Domains

Domains that you create of configure for FCoE support include the following:

Physical Domain

A virtual domain created to support LANs for FCoE VLAN Discovery. The Physical domain will specify the VLAN pool to support FCoE VLAN discovery.

Fibre Channel Domain

A virtual domain created to support virtual SANs for FCoE connections.

A Fibre Channel domain specifies a VSAN pool, VLAN pool and the VSAN Attribute over which the FCoE traffic is carried.

- **VSAN pool** - a set of virtual SANs which you associate with existing VLANs. Individual VSANs can be assigned to associated FCoE-enabled interfaces in the same way that VLANs can be assigned to those interfaces for Ethernet connectivity.
- **VLAN pool** - the set of VLANs available to be associated with individual VSANs.
- **VSAN Attribute** - The mapping of a VSAN to a VLAN.

Tenant Entities

Under the Tenant tab, you configure bridge domain and EPG entities to access the FCoE ports and exchange the FCoE traffic.

The entities include the following:

Bridge Domain (configured for FCoE support)

A bridge domain created and configured under a tenant to carry FCoE traffic for applications that use FCoE connections.

Application EPG

The EPG under the same tenant to be associated with the FCoE bridge domain.

Fibre Channel Path

Specifies the interfaces enabled as FCoE F ports or NP ports to be associated with the selected EPG. After you associate the Fibre Channel path with an EPG the FCoE interface is deployed in the specified VSAN.

Deploying FCoE vFC Ports Using the APIC GUI

The APIC GUI enables you to create customized node policy groups, leaf profiles, interface policy groups, interface profiles, and virtual SAN domains that system administrators can re-use to ensure that all interfaces they designate as F ports or NP ports to handle FCoE traffic have consistent FCoE-related policies applied.

Before you begin

- The ACI fabric is installed.
- If you deploy over a port channel (PC) topology, the port channel is set up as described in Cisco ACI Leaf Switch Port Channel Configuration Using the GUI, on page 82.
- If you deploy over a virtual port channel (vPC) topology, the vPC is set up as described in Configuring a Cisco ACI Leaf Switch Virtual Port Channel Using the Interface Configuration Model Using the GUI, on page 97.

Procedure

Step 1 Create an FCoE supportive switch policy group to specify and combine all the leaf switch policies that support FCoE configuration.

This policy group will be applied to the leaf switches that you want to serve as NPV hosts.

a) In the APIC GUI, starting on the APIC menu bar, click **Fabric** > **Access Policies** > **Switches** > **Leaf Switches** > **Policy Groups**.
b) Right-click **Policy Groups** and click **Create Access Switch Policy Group**.
c) In the **Create Access Switch Policy Group** dialog, specify the settings described below and then click **Submit**.

Policy	Description
Name	Identifies the switch policy group.

Policy	Description
	Enter a name that indicates the FCoE supportive function of this switch policy group. For example, **`fcoe_switch_policy_grp`**.
Fibre Channel SAN Policy	Specifies the following SAN Policy values: • FC Protocol EDTOV (default: 2000) • FC Protocol RATOV (default : 10000) • MAC address prefix (also called FC map) used by the leaf switch. This value should match the value of the peer device connected on the same port. Typically the default value OE:FC:00 is used. Click the drop-down option box. • To use the default EDTOV, RATOV, and MAC address prefix values, click **default**. • To use the value specified in an existing policy, click that policy. • To create a new policy to specify a new customized MAC address prefix, click **Create Fibre Channel SAN Policy** and follow the prompts.

Step 2 Create a leaf profile for leaf switches to support FCoE traffic.

This profile specifies a switch or set of leaf switches to assign the switch policy group that was configured in the previous step. This association enables that set of switches to support FCoE traffic with pre-defined policy settings.

a) Starting at the APIC menu bar, click **Fabric** > **Access Policies** > **Switches** > **Leaf Switches** > **Profiles**
b) Right-click **Leaf Profiles**, then click **Create Leaf Profile**.
c) In the **Create Leaf Profile** dialog create and name the leaf profile (for example: NPV-1)
d) Also in the **Create Leaf Profile** dialog, locate the **Leaf Selectors** table, click +to create a new table row and specify the leaf switches to serve as NPV devices.
e) In the new table row choose a leaf name, blocks, and assign the switch policy group that you created in the previous step.
f) Click **Next** and then click **Finish**.

Step 3 Create at least two FCoE-supportive interface policy groups: one to combine all policies that support FCoE F port interfaces, and one to combine all policies that support FCoE NP port interfaces.

These interface policy groups are to be applied to the interface profiles that are applied to interfaces that are to serve as F ports and NP ports.

a) On the APIC menu bar, click **Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Policy Groups**.
b) Right-click **Policy Groups**, then, depending on how port access is configured, click one of the following options: **Create Leaf Access Port Policy Group**, **Create PC Interface Port Policy**, or **Create vPC Interface Port Policy Group**.

Note
- If you deploy over a PC interface, view Cisco ACI Leaf Switch Port Channel Configuration Using the GUI, on page 82 for additional information.
- If you deploy over a vPC interface, view Configuring a Cisco ACI Leaf Switch Virtual Port Channel Using the Interface Configuration Model Using the GUI, on page 97 for additional information.

c) In the policy group dialog, specify for inclusion the Fibre Channel Interface policy, the slow drain policy, and the priority flow control policy you configure.

Policy	Description
Name	Name of this policy group. Enter a name that indicates the FCoE supportive function of this Leaf Access Port Policy Group and the port type, (F or NP) that it is intended to support, for example: **`fcoe_f_port_policy`** or **`fcoe_np_port_policy`**.
Priority Flow Control Policy	Specifies the state of the Priority Flow Control (PFC) on the interfaces to which this policy group is applied. Options include the following: • **Auto** (the default value) Enables priority flow control (PFC) on local port on the no-drop CoS as configured, on the condition that values advertised by the DCBX and negotiated with the peer succeed. Failure causes priority flow control to be disabled on the no-drop CoS. • **Off** disables FCoE priority flow control on the local port under all circumstances. • **On** enables FCoE PFC on the local port under all circumstances. Click the drop-down option box: • To use the default values, click **default**. • To use the value specified in an existing policy, click that policy. • To create a new policy specifying different values, click **Create Priority Flow Control Policy** and follow the prompts. **Note** PFC requires that Class of Service (CoS) levels be globally enabled for the fabric and assigned to the profiles of applications that generate FCoE traffic. Also CoS Preservation must be enabled. To enable it, navigate to **Fabric** > **Access Policies** > **Policies** > **Global** > **QoS Class** and enable **Preserve COS Dot1p Preserve**.
Slow Drain Policy	Specifies how to handle FCoE packets that are causing traffic congestion on the ACI fabric. Options include the following: • Congestion Clear Action (default: disabled) Action to be taken during FCoE traffic congestion. Options include: • Err - disable - Disable the port. • Log - Record congestion in the Event Log. • Disabled- Take no action. • Congestion Detect Multiplier (default: 10) The number of pause frames received on a port that triggers a congestion clear action to address FCoE traffic congestion. • Flush Admin State

Policy	Description
	• Enabled - Flush the buffer. • Disabled - Don't flush the buffer. • Flush Timeout (default: 500 milliseconds) Threshold in milliseconds to trigger buffer flush drop during congestion. • To use the default values, click **default**. • To use the value specified in an existing policy, click that policy. • To create a new policy specifying different values, click **Create Slow Drain Policy** and follow the prompts.

Step 4 Create at least two interface profiles: one profile to support F port connections, one profile to support NP port connections, and optional additional profiles to be associated with additional port policy variations.

a) Starting at the APIC bar menu click **Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Profiles**.
b) Right-click **Profiles** and choose **Create Leaf Interface Profile**.
c) In the **Create Leaf Interface Profile** dialog, enter a descriptive name for the profile, for example,**`FCoE_F_port_Interface_profile-1`**.
d) Locate the **Interface Selectors** table and click + to display the **Create Access Port Selector** dialog. This dialog enables you to display a range of interfaces and apply settings to the fields described in the following table.

Option	Description
Name	A descriptive name for this port selector.
Interface IDs	Specifies the set of interfaces to which this range applies. • To include all interfaces in the switch, choose **All**. • To include an individual interface in this range, specify single Interface ID, for example: **`1/20`**. • To include a range of interfaces in this range, enter the lower and upper values separated by a dash, for example: **`1/10 - 1/15`**. **Note** Specify separate, non-overlapping ranges of interfaces when configuring interface profiles for F ports and an NP port.
Interface Policy Group	The name of either the F port interface policy group or the NP port policy group that you configured in the previous step. • To designate the interfaces included in this profile as F ports, choose the interface policy group that you configured for F ports. • To designate the interfaces included in the profile as NP ports, choose the interface policy group that you configured for NP ports.

Step 5 Click **Submit**. Repeat the previous step so that you at least have interface profiles for both F ports and an NP port.

Step 6 Configure whether to apply global QoS policies to FCoE traffic.

You can specify different QoS policies to different levels (1, 2, 4, 5, or 6) of FCoE traffic.

a) Starting at the APIC bar menu click **Fabric** > **Access Policies** > **Policies** > **Global** > **QoS Class** and enable the **Preserve CoS** flag in the **QoS Class** pane.

b) In the **QoS Class - Level 1**, **QoS Class - Level 2**, **QoS Class - Level 4**, **QoS Class - Level 5**, or **QoS Class - Level 6** dialog, edit the following fields to specify the PFC and no-drop CoS. Then click **Submit.**

Note
Only 1 Level can be configured for PFC and no-drop CoS.

Policy	Description
PFC Admin State	Whether to enable priority flow control to this level of FCoE traffic (default value is false). Enabling priority flow control sets the **Congestion Algorithm** for this level of FCoE traffic to **no-drop**.
No-Drop-CoS	The CoS level to impose no drop FCoE packet handling even in the case of FCoE traffic congestion.

Step 7 Define a Fibre Channel domain. Create a set of virtual SANs (VSANs) and map them to set of existing VLANs.

a) Starting at the APIC bar menu click **Fabric** > **Access Policies** > **Physical and External Domains** > **Fibre Channel Domains**.

b) Right-click **Fibre Channel Domains** and click **Create Fibre Channel Domain**.

c) In the **Fibre Channel Domain** dialog, specify the following settings:

Option	Description/Action
Name	Specifies the name or label you want to assign the VSAN domain you are creating. (For example: vsan-dom2)
VSAN Pool	The pool of VSANs assigned to this domain. • To select an existing VSAN pool, click the drop-down and choose a listed pool. If you want to revise it, click the Edit icon. • To create a VSAN pool, click **Create a VSAN Pool**. If you open the dialog to create a VSAN pool, follow the prompts configure the following: • A **Static** resource allocation method to support FCoE. • a range of VSANs that will be available to assign to FCoE F port interfaces and NP port interfaces. **Note** Minimum range value is 1. Maximum range value is 4078. Configure multiple ranges of VSANs if necessary.
VLAN Pool	The pool of VLANS available to be mapped to by the members of the VSAN pool.

Option	Description/Action
	A VLAN pool specifies numerical ranges of VLANs you want available to support FCoE connections for this domain. The VLANs in the ranges you specify are available for VSANs to map to them. • To select an existing VLAN pool, click the drop-down and choose a listed pool. If you want to revise it, click the Edit icon. • To create a VLAN pool, click **Create a VLAN Pool**. If you open the dialog to create a VLAN pool, follow the prompts configure the following: • A **Static** resource allocation method to support FCoE. • a range of VLANs that will be available for VSANs to map to. **Note** Minimum range value is 1. Maximum range value is 4094. Configure multiple ranges of VLANs if necessary.
VSAN Attr	The VSAN Attributes map for this domain. The VSAN Attributes map VSANs in the VSAN pool to VLANs in the VLAN pool. • To select an existing VSAN Attributes map, click the drop-down and choose a listed map. If you want to revise it, click the Edit icon. • To create a VSAN Attributes map, click **Create VSAN Attributes**. If you open the dialog to configure the VSAN attributes, follow the prompts configure the following: • The appropriate load balancing option (src-dst-ox-id or src-dst-id). • Mapping of individual VSANs to individual VLANs, for example: vsan-8 to vlan-10 **Note** Only VSANs and VLANs in the ranges you specified for this domain can be mapped to each other.

Step 8 Create an attached entity profile to bind the Fibre Channel domain with the interface policy group.

a) On the APIC menu bar, click **Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Policy Groups** > ***interface_policy_group_name***.

In this step *interface_policy_group_name* is the interface policy group that you defined in Step 3.

b) In the interface policy group dialog, Click the Attached Entity Profile drop-down and choose an existing Attached Entity Profile or click **Create Attached Entity Profile** to create a new one.

c) In the Attached Entity Profile dialog specify the following settings:

Field	Description
Name	A name for this Attached Entity Profile.
Domains To Be Associated To Interfaces	Lists the domain to be associated with the interface policy group. In this case, choose the Fibre Channel domain you configured in Step 7.

Field	Description
	Click **Submit**.

Step 9 Associate the leaf profile and the F port and NP port interface profiles.

a) Starting at the APIC menu bar, click **Fabric** > **Access Policies** > **Switches** > **Leaf Switches** > **Profiles** then click the name of the leaf profile you configured in Step 2.

b) In the **Create Leaf Profile** dialog, locate the **Associated Interface Selector Profiles** table, click +to create a new table row and choose the F port interface profile you created in Step 4.

c) Again on the **Associated Interface Selector Profiles** table, click +to create a new table row and choose the NP port interface profile you created in Step 4.

d) Click **Submit**.

What to do next

After successful deployment of virtual F ports and NP ports to interfaces on the ACI fabric, the next step is for system administrators to enable EPG access and connection over those interfaces.

For more information, see Deploying EPG Access to vFC Ports Using the APIC GUI, on page 135.

Deploying EPG Access to vFC Ports Using the APIC GUI

After you have configured ACI fabric entities to support FCoE traffic and F port and NP port functioning of designated interfaces, your next step is to configure EPG access to those ports.

Before you begin

- The ACI fabric is installed.
- A Fibre Channel Forwarding (FCF) switch, connected to a FC network (for example, SAN storage), is physically attached by Ethernet to an ACI leaf switch port.
- A host application that needs to access the FC network is physically attached by Ethernet to a port on the same ACI leaf switch.
- Leaf policy groups, leaf profiles, interface policy groups, interface profiles, and Fibre Channel domains have all been configured to support FCoE traffic.

Procedure

Step 1 Under an appropriate tenant configure an existing bridge domain to support FCoE or create a bridge domain to support FCoE.

Option:	Actions
To configure an existing bridge domain for FCoE	**a.** Click **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > ***bridge_domain_name***. **b.** In the **Type** field of the bridge domain's **Properties** panel, click **fc**.

Option:	Actions
	c. Click **Submit**.
To create a new bridge domain for FCoE	**a.** Click **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > **Actions** > **Create a Bridge Domain**. **b.** In the **Name** field of the **Specify Bridge Domain for the VRF** dialog, enter a bridge domain name. **c.** In the **Type** field of **Specify Bridge Domain for the VRF dialog**, click **fc**. **d.** In VRF field select a VRF from the drop-down or click **Create VRF** to create and configure a new VRF. **e.** Finish the bridge domain configuration. **f.** Click **Submit**.

Step 2 Under the same tenant, configure an existing EPG or create a new EPG to associate with the FCoE-configured bridge domain.

Option:	Actions
To associate an existing EPG	**a.** Click **Tenant** > **<*tenant_name*>** > **Application Profiles** > **<*application_profile_name*>** > **Application EPGs** > **<*epg_name*>**. **b.** In the **QoS class** field choose the quality of service (**Level1**, **Level2**, **Level4**, **Level5**, or **Level6**) to assign to traffic generated by this EPG. If you configured one of the QoS levels for priority-flow control no-drop congestion handling and you want FCoE traffic handled with no-dropped packet priority, assign that QoS level to this EPG. **c.** In the **Bridge Domain** field of the EPG's **Properties** panel, click the drop-down list and choose the name of a bridge domain configured for Type: fcoe. **d.** Click **Submit**. **Note** If you change the **Bridge Domain** field, you must wait 30-35 seconds between changes. Changing the Bridge Domain field too rapidly causes vFC interfaces on the NPV Switch to fail and a switch reload must be executed.
To create and associate a new EPG	**a.** Click **Tenant** > **<*tenant_name*>** > **Application Profiles** > **<*application_profile_name*>** > **Application EPGs**. **b.** Right-click **Application EPGs** and click **Create Application EPG**. **c.** In the **QoS class** field choose the quality of service (**Level1**, **Level2**, **Level4**, **Level5**, or **Level6**) to assign to traffic generated by this EPG. If you configured one of the QoS levels for priority-flow control no-drop congestion handling and you want FCoE traffic handled with no-dropped packet priority, assign that QoS level to this EPG.

Option:	Actions
	d. In the **Bridge Domain** field of the **Specify the EPG Identity** dialog, click the drop-down list and choose the name of a bridge domain configured for Type: fcoe. **Note** If you change the **Bridge Domain** field, you must wait 30-35 seconds between changes. Changing the Bridge Domain field too rapidly causes vFC interfaces on the NPV Switch to fail and a switch reload must be executed. **e.** Finish the bridge domain configuration. **f.** Click **Finish**.

Step 3 Add a Fibre Channel Domain association with the EPG.

a) Click **Tenant** > **<*tenant_name*>** > **Application Profiles** > **<*application_profile_name*>** > **Application EPGs** > **<*epg_name*>** > **Domains (VMs and Bare Metal)**.
b) Right-click **Domains (VMs and Bare Metal)** and click **Add Fibre Channel Domain Association**.
c) In the **Add Fibre Channel Domain Association** dialog, locate the **Fibre Channel Domain Profile Field.**
d) Click the drop-down list and choose the name of the Fibre Channel domain that you previously configured.
e) Click **Submit**.

Step 4 Under the associated EPG define a Fibre Channel path.

The Fibre Channel path specifies the interfaces enabled as FCoE F ports or NP ports to be associated with the selected EPG.

a) Click **Tenant** > **<*tenant_name*>** > **Application Profiles** > **<*application_profile_name*>** > **Application EPGs** > **<*epg_name*>** > **Fibre Channel (Paths)**.
b) Right-click **Fibre Channel (Paths)** and click **Deploy Fibre Channel**.
c) In the **Deploy Fibre Channel** dialog configure the following settings:

Option:	Actions
Path Type	The type of interface (Port, Direct Port Channel, or Virtual Port Channel) being accessed for sending and receiving FCoE traffic.
Path	The Node-interface path through which FCoE traffic associated with the selected EPG will flow. Click the drop-down list and choose from the listed interfaces. . **Note** Choose only the interfaces previously configured as F ports or NP ports. Choosing interfaces that you did not configure causes only default values to apply to those interfaces. **Note** To deploy FCoE over FEX, select the FEX ports previously configured.
VSAN	The VSAN which will use the interface selected in the **Path** field. **Note** The specified VSAN must be in the range of VSANs that was designated for the VSAN pool. In most cases, all interfaces that this EPG is configured to access must be assigned the same VSAN, unless you specify a Fibre Channel path over a Virtual Port Channel (VPC) connection. In that case, you can specify two VSANs, one for each leg of the connection.

Option:	Actions
VSAN Mode	The mode (**Native** or **Regular**) in which the selected VSAN accesses the selected interface. Every interface configured for FCoE support, requires one VSAN and only one VSAN configured for Native mode. Any additional VSANs assigned to the same interface must access it in Regular mode.
Pinning label	(Optional) This option applies only if you are mapping access to an F port and it is necessary to bind this F port with a specific uplink NP port. It associates a pinning label (pinning label 1 or pinning label 2) with a specific NP port. You can then assign that pinning label to the target F port. This association causes the associated NP port to serve in all cases as the uplink port to the target F Port. Choose a pinning label and associate it with an interface configured as an NP port. This option implements what is also referred to as "traffic-mapping." **Note** The F port and the associated Pinning Label NP port must be on the same Leaf switch.

Step 5 Click **Submit**.

Step 6 Repeat Steps 4 and 5 for every FCoE enabled interface to which you are mapping EPG access.

Step 7 Verify successful deployment, as follows:

a) Click **Fabric** > **Inventory** > ***Pod_name*** > ***leaf_name*** > **Interfaces** > **VFC interfaces**.

The interfaces on which you deployed ports are listed under VFC Interfaces.

What to do next

After you have set up EPG access to the vFC interfaces, the final step is to set up the network supporting the FCoE initialization protocol (FIP), which enables discovery of those interfaces.

For more information, see Deploying the EPG to Support the FCoE Initiation Protocol, on page 138.

Deploying the EPG to Support the FCoE Initiation Protocol

After you have configured FCoE EPG access to your server ports, you must also configure EPG access to support the FCoE Initiation Protocol (FIP).

Before you begin

- The ACI fabric is installed.
- A host application that needs to access the FC network is physically attached by Ethernet to a port on the same ACI leaf switch.
- Leaf policy groups, leaf profiles, interface policy groups, interface profiles, and Fibre Channel domains have all been configured to support FCoE traffic as described in the topic Deploying EPG Access to vFC Ports Using the APIC GUI, on page 135.
- EPG access to the vFC ports is enabled as described in the topic Deploying EPG Access to vFC Ports Using the APIC GUI, on page 135.

Procedure

Step 1 Under the same tenant configure an existing bridge domain to support FIP or create a regular bridge domain to support FIP.

Option:	Actions
To configure an existing bridge domain for FCoE	**a.** Click **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > ***bridge_domain_name***. **b.** In the **Type** field of the bridge domain's **Properties** panel, click **Regular**. **c.** Click **Submit**.
To create a new bridge domain for FCoE	**a.** Click **Tenant** > ***tenant_name*** > **Networking** > **Bridge Domains** > **Actions** > **Create a Bridge Domain**. **b.** In the **Name** field of the **Specify Bridge Domain for the VRF** dialog, enter a bridge domain name. **c.** In the **Type** field of **Specify Bridge Domain for the VRF dialog**, click **Regular**. **d.** In VRF field select a VRF from the drop-down or click **Create VRF** to create and configure a new VRF. **e.** Finish the bridge domain configuration. **f.** Click **Submit**.

Step 2 Under the same tenant, configure an existing EPG or create a new EPG to associate with the regular-type bridge domain.

Option:	Actions
To associate an existing EPG	**a.** Click **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs** > ***epg_name***. **b.** In the **Bridge Domain** field of the EPG's **Properties** panel, click the drop-down list and choose the name of the regular bridge domain that you just configured to support FIP. **c.** Click **Submit**.
To create and associate a new EPG	**a.** Click **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs**. **b.** Right-click **Application EPGs** and click **Create Application EPG**. **c.** In the **Bridge Domain** field of the **Specify the EPG Identity** dialog, click the drop-down list and choose the name of the regular bridge domain that you just configured to support FIP. **d.** Finish the bridge domain configuration. **e.** Click **Finish**.

Step 3 Add a Physical Domain association with the EPG.

a) Click **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs** > ***epg_name*** > **Domains & Bare Metal**.
b) Right-click **Domains & Bare Metal** and click **Add Physical Domain Association**.
c) In the **Add Physical Domain Association** dialog, locate Physical Domain Profile Field.
d) Click the drop-down list and choose the name of the physical domain that contains the LAN that intended for use in FIP support.
e) Click **Submit**.

Step 4 Under the associated EPG define a path.

The path specifies the interfaces enabled as FCoE F ports or NP ports to be associated with the selected EPG.

a) Click **Tenant** > ***tenant_name*** > **Application Profiles** > **ap1** > **Application EPGs** > ***epg_name*** > **Static Ports**.
b) Right-click **Static Ports** and click **Deploy Static EPG on PC, VPC, or Interface**.
c) In the **Path Type** field, specify the port type (Port, Direct Port Channel, or Virtual Port Channel) on which you want to deploy an F mode vFC.
d) In the **Path** field, specify all the paths on which are deployed the F ports.
e) Choose the VLAN Encap that you want to use as your FCoE VLAN discovery and 802.1p(access) as port mode.
f) Click **Submit**.

The FCoE components will begin the discovery process to initiate the operation of the FCoE network.

Undeploying FCoE Connectivity Using the APIC GUI

To undo FCoE enablement of leaf switch interfaces on the ACI fabric, delete the Fibre Channel path and Fibre Channel domain and its elements that you defined in Deploying FCoE vFC Ports Using the APIC GUI, on page 129.

Note If during clean up you delete the Ethernet configuration object (infraHPortS) for a vFC port (for example, in the **Interface Selector** table on the **Leaf Interface Profiles** page of the GUI), the default vFC properties remain associated with that interface. For example it the interface configuration for vFC NP port 1/20 is deleted, that port remains a vFC port but with default F port setting rather than non-default NP port setting applied.

Before you begin

You must know the name of the Fibre Channel path and Fibre Channel domain including its associated VSAN pool, VLAN pool, and VSAN Attributes map that you specified during FCoE deployment.

Procedure

Step 1 Delete the associated Fibre Channel path to undeploy vFC from the port/vsan whose path was specified on this deployment.

This action removes vFC deployment from the port/vsan whose path was specified on this deployment.

a) Click **Tenants** > ***tenant_name*** > **Application Profiles** > ***app_profile_name*** > **Application EPGs** > ***app_epg_name*** > **Fibre Channel (Paths)**. Then right-click the name of the target Fibre Channel path and choose **Delete**.
b) Click **Yes** to confirm the deletion.

Step 2 Delete the VLAN to VSAN map that you configured when you defined the Fibre Channel domain.

This action removes vFC deployment from all the elements defined in the map.

a) Click **Fabric** > **Access Policies** > **Pools** > **VSAN Attributes**. Then right-click the name of the target map and choose **Delete**.
b) Click **Yes** to confirm the deletion.

Step 3 Delete the VLAN and VSAN pools that you defined when you defined the Fibre Channel domain.

This action eliminates all vFC deployment from the ACI fabric.

a) Click **Fabric** > **Access Policies** > **Pools** > **VSAN** and then, right-click the name of the target VSAN pool name and choose **Delete**.
b) Click **Yes** to confirm the deletion.
c) Click **Fabric** > **Access Policies** > **Pools** > **VLAN** then, right-click the target VLAN pool name and choose **Delete**.
d) Click **Yes** to confirm the deletion.

Step 4 Delete the Fibre Channel Domain that contained the VSAN pool, VLAN pool, and Map elements you just deleted.

a) Click **Tenants** > **tenant_name** > **Application Profiles** > **Fibre Channel Domains**. Then right-click the name of the target Fibre Channel Domain and choose **Delete**.
b) Click **Yes** to confirm the deletion.

Step 5 You can delete the tenant/EPG/App and the selectors if you don't need them.

Option	Action
If you want to delete the associated application EPG but save the associated tenant and application profile:	Click **Tenants** > ***tenant_name*** > **Application Profiles** > ***app_profile_name*** > **Application EPGs**, right-click the name of the target application EPG, choose **Delete**, then click **Yes** to confirm deletion.
If you want to delete the associated application profile but save the associated tenant:	Click **Tenants** > ***tenant_name*** > **Application Profiles**, right-click the name of the target application profile, choose **Delete**, then click **Yes** to confirm deletion.
If you want to delete the associated tenant:	Click **Tenants** > , right-click the name of the target tenant, choose **Delete**, then click **Yes** to confirm deletion.

Configuring FCoE Using the NX_OS Style CLI

FCoE NX-OS Style CLI Configuration

Configuring FCoE Connectivity Without Policies or Profiles Using the NX-OS Style CLI

The following sample NX-OS style CLI sequences configure FCoE connectivity for EPG **e1** under tenant **t1** without configuring or applying switch-level and interface-level policies and profiles.

Procedure

	Command or Action	Purpose

Step 1	Under the target tenant configure a bridge domain to support FCoE traffic.
	Example:
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# vrf context v1`
	`apic1(config-tenant-vrf)# exit`
	`apic1(config-tenant)# bridge-domain b1`
	`apic1(config-tenant-bd)# fc`
	`apic1(config-tenant-bd)# vrf member v1`
	`apic1(config-tenant-bd)# exit`
	`apic1(config-tenant)# exit`
		The sample command sequence creates bridge domain **b1** under tenant **t1** configured to support FCoE connectivity.

Step 2	Under the same tenant, associate the target EPG with the FCoE-configured bridge domain.
	Example:
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# application a1`
	`apic1(config-tenant-app)# epg e1`
	`apic1(config-tenant-app-epg)# bridge-domain member b1`
	`apic1(config-tenant-app-epg)# exit`
	`apic1(config-tenant-app)# exit`
	`apic1(config-tenant)# exit`
		The sample command sequence creates EPG **e1** and associates that EPG with the FCoE-configured bridge domain **b1**.

Step 3	Create a VSAN domain, VSAN pools, VLAN pools and VSAN to VLAN mapping.
	Example:
	A
	`apic1(config)# vsan-domain dom1`
	`apic1(config-vsan)# vsan 1-10`
	`apic1(config-vsan)# vlan 1-10`
	`apic1(config-vsan)# fcoe vsan 1 vlan 1 loadbalancing src-dst-ox-id`
	`apic1(config-vsan)# fcoe vsan 2 vlan 2`
	Example:
	B
		In **Example A**, the sample command sequence creates VSAN domain, **dom1** with VSAN pools and VLAN pools, maps VSAN 1 to VLAN 1 and maps VSAN 2 to VLAN 2
		In **Example B**, an alternate sample command sequence creates a reusable VSAN attribute template **pol1** and then creates VSAN domain **dom1**, which inherits the attributes and mappings from that template.

	Command or Action	Purpose
	apic1(config)# template vsan-attribute pol1 apic1(config-vsan-attr)# fcoe vsan 2 vlan 12 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# fcoe vsan 3 vlan 13 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# exit apic1(config)# vsan-domain dom1 apic1(config-vsan)# vsan 1-10 apic1(config-vsan)# vlan 1-10 apic1(config-vsan)# inherit vsan-attribute pol1 apic1(config-vsan)# exit	
Step 4	Create the physical domain to support the FCoE Initialization (FIP) process. **Example:** apic1(config)# vlan-domain fipVlanDom apic1(config-vlan)# vlan 120 apic1(config-vlan)# exit	In the example, the command sequence creates a regular VLAN domain, **fipVlanDom**, which includes VLAN **120** to support the FIP process.
Step 5	Under the target tenant configure a regular bridge domain. **Example:** apic1(config)# tenant t1 apic1(config-tenant)# vrf context v2 apic1(config-tenant-vrf)# exit apic1(config-tenant)# bridge-domain fip-bd apic1(config-tenant-bd)# vrf member v2 apic1(config-tenant-bd)# exit apic1(config-tenant)# exit	In the example, the command sequence creates bridge domain **fip-bd**.
Step 6	Under the same tenant, associate this EPG with the configured regular bridge domain. **Example:** apic1(config)# tenant t1 apic1(config-tenant)# application a1 apic1(config-tenant-app)# epg epg-fip apic1(config-tenant-app-epg)# bridge-domain member fip-bd apic1(config-tenant-app-epg)# exit apic1(config-tenant-app)# exit apic1(config-tenant)# exit	In the example, the command sequence associates EPG **epg-fip** with bridge domain **fip-bd**.
Step 7	Configure a VFC interface with F mode. **Example:** **A** apic1(config)# leaf 101 apic1(config-leaf)# interface ethernet 1/2 apic1(config-leaf-if)# vlan-domain member fipVlanDom apic1(config-leaf-if)# switchport trunk native vlan 120 tenant t1 application a1 epg epg-fip apic1(config-leaf-if)# exit apic1(config-leaf)# exit apic1(config-leaf)# interface vfc 1/2	In example **A** the command sequence enables interface **1/2** on leaf switch **101** to function as an **F** port and associates that interface with VSAN domain **dom1**. Each of the targeted interfaces must be assigned one (and only one) VSAN in native mode. Each interface may be assigned one or more additional VSANs in regular mode. The sample command sequence associates the target interface **1/2** with: • VLAN **120** for FIP discovery and associates it with EPG **epg-fip** and application **a1** under tenant **t1**.

	Command or Action	Purpose
	apic1(config-leaf-if)# switchport mode f apic1(config-leaf-if)# vsan-domain member dom1 apic1(config-leaf-if)# switchport vsan 2 tenant t1 application a1 epg e1 apic1(config-leaf-if)# switchport trunk allowed vsan 3 tenant t1 application a1 epg e2 apic1(config-leaf-if)# exit **Example:** **B** apic1(config)# vpc context leaf 101 102 apic1(config-vpc)# interface vpc vpc1 apic1(config-vpc-if)# vlan-domain member vfdom100 apic1(config-vpc-if)# vsan-domain member dom1 apic1(config-vpc-if)# #For FIP discovery apic1(config-vpc-if)# switchport trunk native vlan 120 tenant t1 application a1 epg epg-fip apic1(config-vpc-if)# switchport vsan 2 tenant t1 application a1 epg e1 apic1(config-vpc-if)# exit apic1(config-vpc)# exit apic1(config)# leaf 101-102 apic1(config-leaf)# interface ethernet 1/3 apic1(config-leaf-if)# channel-group vpc1 vpc apic1(config-leaf-if)# exit apic1(config-leaf)# exit **Example:** **C** apic1(config)# leaf 101 apic1(config-leaf)# interface vfc-po pc1 apic1(config-leaf-if)# vsan-domain member dom1 apic1(config-leaf-if)# switchport vsan 2 tenant t1 application a1 epg e1 apic1(config-leaf-if)# exit apic1(config-leaf)# interface ethernet 1/2 apic1(config-leaf-if)# channel-group pc1 apic1(config-leaf-if)# exit apic1(config-leaf)# exit	• VSAN **2** as a native VSAN and associates it with EPG **e1** and application **a1** under tenant **t1**. • VSAN **3** as a regular VSAN. In example **B**, the command sequence configures a vFC over a vPC with the same VSAN on both the legs. From the CLI you cannot specify different VSANs on each log. The alternate configuration can be carried out in the APIC advanced GUI.
Step 8	Configure a VFC interface with NP mode. **Example:** apic1(config)# leaf 101 apic1(config-leaf)# interface vfc 1/4 apic1(config-leaf-if)# switchport mode np apic1(config-leaf-if)# vsan-domain member dom1	The sample command sequence enables interface **1/4** on leaf switch **101** to function as an **NP** port and associates that interface with VSAN domain **dom1**.
Step 9	Assign the targeted FCoE-enabled interfaces a VSAN. **Example:** apic1(config-leaf-if)# switchport trunk allowed vsan 1 tenant t1 application a1 epg e1 apic1(config-leaf-if)# switchport vsan 2 tenant t4 application a4 epg e4	Each of the targeted interfaces must be assigned one (and only one) VSAN in native mode. Each interface may be assigned one or more additional VSANs in regular mode. The sample command sequence assigns the target interface to VSAN **1** and associates it with EPG **e1** and application **a1** under tenant **t1**. "trunk allowed" assigns vsan 1 regular mode status. The command sequence also assigns the interface a required **native mode** VSAN **2**. As this example

	Command or Action	Purpose
		shows, it is permissible for different VSANs to provide different EPGs running under different tenants access to the same interfaces.

Configuring FCoE Connectivity With Policies and Profiles Using the NX-OS Style CLI

The following sample NX-OS style CLI sequences create and use policies to configure FCoE connectivity for EPG **e1** under tenant **t1**.

Procedure

	Command or Action	Purpose

Step 1	Under the target tenant configure a bridge domain to support FCoE traffic.
	Example:
	`apic1# configure`
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# vrf context v1`
	`apic1(config-tenant-vrf)# exit`
	`apic1(config-tenant)# bridge-domain b1`
	`apic1(config-tenant-bd)# fc`
	`apic1(config-tenant-bd)# vrf member v1`
	`apic1(config-tenant-bd)# exit`
	`apic1(config-tenant)# exit`
	`apic1(config)#`
		The sample command sequence creates bridge domain **b1** under tenant **t1** configured to support FCoE connectivity.

Step 2	Under the same tenant, associate your target EPG with the FCoE configured bridge domain.
	Example:
	`apic1(config)# tenant t1`
	`apic1(config-tenant)# application a1`
	`apic1(config-tenant-app)# epg e1`
	`apic1(config-tenant-app-epg)# bridge-domain member b1`
	`apic1(config-tenant-app-epg)# exit`
	`apic1(config-tenant-app)# exit`
	`apic1(config-tenant)# exit`
	`apic1(config)#`
		The sample command sequence creates EPG **e1** associates that EPG with FCoE-configured bridge domain **b1**.

Step 3	Create a VSAN domain, VSAN pools, VLAN pools and VSAN to VLAN mapping.
	Example:
	A
	`apic1(config)# vsan-domain dom1`
	`apic1(config-vsan)# vsan 1-10`
	`apic1(config-vsan)# vlan 1-10`
	`apic1(config-vsan)# fcoe vsan 1 vlan 1 loadbalancing`
		In **Example A**, the sample command sequence creates VSAN domain, **dom1** with VSAN pools and VLAN pools, maps VSAN 1 VLAN 1 and maps VSAN 2 to VLAN 2
		In **Example B**, an alternate sample command sequence creates a reusable vsan attribute template **pol1** and then creates VSAN domain **dom1**, which inherits the attributes and mappings from that template.

	Command or Action	Purpose
	src-dst-ox-id apic1(config-vsan)# fcoe vsan 2 vlan 2 **Example:** **B** apic1(config)# template vsan-attribute pol1 apic1(config-vsan-attr)# fcoe vsan 2 vlan 12 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# fcoe vsan 3 vlan 13 loadbalancing src-dst-ox-id apic1(config-vsan-attr)# exit apic1(config)# vsan-domain dom1 apic1(config-vsan)# inherit vsan-attribute pol1 apic1(config-vsan)# exit	
Step 4	Create the physical domain to support the FCoE Initialization (FIP) process. **Example:** apic1(config)# vlan-domain fipVlanDom apic1(config)# vlan-pool fipVlanPool	
Step 5	Configure a Fibre Channel SAN policy. **Example:** apic1# apic1# configure apic1(config)# template fc-fabric-policy ffp1 apic1(config-fc-fabric-policy)# fctimer e-d-tov 1111 apic1(config-fc-fabric-policy)# fctimer r-a-tov 2222 apic1(config-fc-fabric-policy)# fcoe fcmap 0E:FC:01 apic1(config-fc-fabric-policy)# exit	The sample command sequence creates Fibre Channel SAN policy **ffp1** to specify a combination of error-detect timeout values (EDTOV), resource allocation timeout values (RATOV), and the default FC map values for FCoE-enabled interfaces on a target leaf switch.
Step 6	Create a Fibre Channel node policy. **Example:** apic1(config)# template fc-leaf-policy flp1 apic1(config-fc-leaf-policy)# fcoe fka-adv-period 44 apic1(config-fc-leaf-policy)# exit	The sample command sequence creates Fibre Channel node policy **flp1** to specify a combination of disruptive load-balancing enablement and FIP keep-alive values. These values also apply to all the FCoE-enabled interfaces on a target leaf switch.
Step 7	Create Node Policy Group. **Example:** apic1(config)# template leaf-policy-group lpg1 apic1(config-leaf-policy-group)# inherit fc-fabric-policy ffp1 apic1(config-leaf-policy-group)# inherit fc-leaf-policy flp1 apic1(config-leaf-policy-group)# exit apic1(config)# exit apic1#	The sample command sequence creates a Node Policy group, **lpg1**, which combines the values of the Fibre Channel SAN policy **ffp1** and Fibre Channel node policy, **flp1**. The combined values of this node policy group can be applied to Node profiles configured later.

	Command or Action	Purpose

Step 8	Create a Node Profile.
	Example:
	`apic1(config)# leaf-profile lp1`
	`apic1(config-leaf-profile)# leaf-group lg1`
	`apic1(config-leaf-group)# leaf 101`
	`apic1(config-leaf-group)# leaf-policy-group lpg1`
		The sample command sequence creates node profile **lp1** associates it with node policy group **lpg1**, node group **lg1**, and leaf switch **101**.

Step 9	Create an interface policy group for F port interfaces.
	Example:
	`apic1(config)# template policy-group ipg1`
	`apic1(config-pol-grp-if)# priority-flow-control mode auto`
	`apic1(config-pol-grp-if)# switchport mode f`
	`apic1(config-pol-grp-if)# slow-drain pause timeout 111`
	`apic1(config-pol-grp-if)# slow-drain congestion-timeout count 55`
	`apic1(config-pol-grp-if)# slow-drain congestion-timeout action log`
		The sample command sequence creates interface policy group **ipg1** and assigns a combination of values that determine priority flow control enablement, F port enablement, and slow-drain policy values for any interface that this policy group is applied to.

Step 10	Create an interface policy group for NP port interfaces.
	Example:
	`apic1(config)# template policy-group ipg2`
	`apic1(config-pol-grp-if)# priority-flow-control mode auto`
	`apic1(config-pol-grp-if)# switchport mode np`
	`apic1(config-pol-grp-if)# slow-drain pause timeout 111`
	`apic1(config-pol-grp-if)# slow-drain congestion-timeout count 55`
	`apic1(config-pol-grp-if)# slow-drain congestion-timeout action log`
		The sample command sequence creates interface policy group **ipg2** and assigns a combination of values that determine priority flow control enablement, NP port enablement, and slow-drain policy values for any interface that this policy group is applied to.

Step 11	Create an interface profile for F port interfaces.
	Example:
	`apic1# configure`
	`apic1(config)# leaf-interface-profile lip1`
	`apic1(config-leaf-if-profile)# description 'test description lip1'`
	`apic1(config-leaf-if-profile)# leaf-interface-group lig1`
	`apic1(config-leaf-if-group)# description 'test description lig1'`
	`apic1(config-leaf-if-group)# policy-group ipg1`
	`apic1(config-leaf-if-group)# interface ethernet 1/2-6, 1/9-13`
		The sample command sequence creates an interface profile **lip1** for F port interfaces, associates the profile with F port specific interface policy group **ipg1**, and specifies the interfaces to which this profile and its associated policies applies.

Step 12	Create an interface profile for NP port interfaces.
	Example:
	`apic1# configure`
	`apic1(config)# leaf-interface-profile lip2`
	`apic1(config-leaf-if-profile)# description 'test description lip2'`
		The sample command sequence creates an interface profile **lip2** for NP port interfaces, associates the profile with NP port specific interface policy group **ipg2**, and specifies the interface to which this profile and its associated policies applies.

	Command or Action	Purpose
	apic1(config-leaf-if-profile)# leaf-interface-group lig2 apic1(config-leaf-if-group)# description 'test description lig2' apic1(config-leaf-if-group)# policy-group ipg2 apic1(config-leaf-if-group)# interface ethernet 1/14	
Step 13	Configure QoS Class Policy for Level 1. **Example:** apic1(config)# qos parameters level1 apic1(config-qos)# pause no-drop cos 3	The sample command sequence specifies the QoS level of FCoE traffic to which priority flow control policy might be applied and pauses no-drop packet handling for Class of Service level 3.

Configuring FCoE Over FEX Using NX-OS Style CLI

FEX ports are configured as port VSANs.

Procedure

Step 1 Configure Tenant and VSAN domain:

Example:

```
apic1# configure
apic1(config)#  tenant t1
apic1(config-tenant)#   vrf context v1
apic1(config-tenant-vrf)#     exit
apic1(config-tenant)#   bridge-domain b1
apic1(config-tenant-bd)#       fc
apic1(config-tenant-bd)#       vrf member v1
apic1(config-tenant-bd)#       exit
apic1(config-tenant)#   application a1
apic1(config-tenant-app)#      epg e1
apic1(config-tenant-app-epg)# bridge-domain member b1
apic1(config-tenant-app-epg)# exit
apic1(config-tenant-app)#      exit
apic1(config-tenant)# exit

apic1(config)#  vsan-domain dom1
apic1(config-vsan)# vlan 1-100
apic1(config-vsan)# vsan 1-100
apic1(config-vsan)# fcoe vsan 2 vlan 2 loadbalancing src-dst-ox-id
apic1(config-vsan)# fcoe vsan 3 vlan 3 loadbalancing src-dst-ox-id
apic1(config-vsan)# fcoe vsan 5 vlan 5
apic1(config-vsan)# exit
```

Step 2 Associate FEX to an interface:

Example:

```
apic1(config)#  leaf 101
apic1(config-leaf)#  interface ethernet 1/12
apic1(config-leaf-if)#  fex associate 111
apic1(config-leaf-if)#  exit
```

Step 3 Configure FCoE over FEX per port, port-channel, and VPC:

Example:

```
apic1(config-leaf)#  interface vfc 111/1/2
apic1(config-leaf-if)#  vsan-domain member dom1
apic1(config-leaf-if)#  switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-leaf-if)# exit

apic1(config-leaf)#  interface vfc-po pc1 fex 111
apic1(config-leaf-if)# vsan-domain member dom1
apic1(config-leaf-if)# switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 111/1/3
apic1(config-leaf-if)# channel-group pc1
apic1(config-leaf-if# exit
apic1(config-leaf)# exit

apic1(config)# vpc domain explicit 12 leaf 101 102
apic1(config-vpc)# exit
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)#     interface vpc vpc1 fex 111 111
apic1(config-vpc-if)#       vsan-domain member dom1
apic1(config-vpc-if)#       switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-vpc-if)#       exit
apic1(config-vpc)#     exit
apic1(config)# leaf 101-102
apic1(config-leaf)# interface ethernet 1/2
apic1(config-leaf-if)# fex associate 111
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 111/1/2
apic1(config-leaf-if)# channel-group vpc1 vpc
apic1(config-leaf-if)# exit
```

Step 4 Verify the configuration with the following command:

Example:

```
apic1(config-vpc)# show vsan-domain detail
vsan-domain : dom1

vsan : 1-100

vlan : 1-100

Leaf          Interface         Vsan  Vlan  Vsan-Mode    Port-Mode  Usage               Operational
 State
------------  ----------------  ----  ----  -----------  ---------
---------------------------------------
101           vfc111/1/2       2       2    Native                  Tenant: t1           Deployed
                                                                     App: a1
                                                                     Epg: e1

101           PC:pc1           5       5    Native                Tenant: t1             Deployed
                                                                     App: a1
                                                                     Epg: e1

101           vfc111/1/3       3       3    Native       F          Tenant: t1           Deployed
                                                                     App: a1
                                                                     Epg: e1
```

Verifying FCoE Configuration Using the NX-OS Style CLI

The following **show** command verifies the FCoE configuration on your leaf switch ports.

Procedure

Use the **show vsan-domain** command to verify FCoE is enabled on the target switch.

The command example confirms FCoE enabled on the listed leaf switches and its FCF connection details.

Example:

```
ifav-isim8-ifc1# show vsan-domain detail
vsan-domain : iPostfcoeDomP1

vsan : 1-20  51-52  100-102  104-110  200  1999  3100-3101  3133
      2000

vlan : 1-20  51-52  100-102  104-110  200  1999  3100-3101  3133
      2000

                                    Vsan    Port                    Operational
 Leaf  Interface       Vsan  Vlan   Mode    Mode  Usage             State
 ----  ---------       ----  ----  ------- ----  ----------------  ------------
 101   vfc1/11         1     1     Regular  F    Tenant: iPost101  Deployed
                                                 App: iPost1
                                                 Epg: iPost1


 101   vfc1/12         1     1     Regular  NP   Tenant: iPost101  Deployed
                                                 App: iPost1
                                                 Epg: iPost1


 101   PC:infraAccBndl 4     4     Regular  NP   Tenant: iPost101  Deployed
         Grp_pc01                                App: iPost4
                                                 Epg: iPost4


 101   vfc1/30         2000        Native        Tenant: t1        Not deployed
                                                 App: a1           (invalid-path)

                                                 Epg: e1
```

Undeploying FCoE Elements Using the NX-OS Style CLI

Any move to undeploy FCoE connectivity from the ACI fabric requires that you remove the FCoE components on several levels.

Procedure

Step 1 List the attributes of the leaf port interface, set its mode setting to default, and then remove its EPG deployment and domain association.

The example sets the port mode setting of interface vfc **1/2** to default and then removes the deployment of EPG **e1** and the association with VSAN Domain **dom1** from that interface.

Example:

```
apic1(config)# leaf 101
apic1(config-leaf)# interface vfc 1/2
apic1(config-leaf-if)# show run
# Command: show running-config leaf 101 interface vfc 1 / 2
# Time: Tue Jul 26 09:41:11 2016
  leaf 101
    interface vfc 1/2
      vsan-domain member dom1
      switchport vsan 2 tenant t1 application a1 epg e1
      exit
    exit
apic1(config-leaf-if)# no switchport mode
apic1(config-leaf-if)# no switchport vsan 2 tenant t1 application a1 epg e1
apic1(config-leaf-if)# no vsan-domain member dom1
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

Step 2 List and remove the VSAN/VLAN mapping and the VLAN and VSAN pools.

The example removes the VSAN/VLAN mapping for **vsan 2**, VLAN pool **1-10**, and VSAN pool **1-10** from VSAN domain **dom1**.

Example:

```
apic1(config)# vsan-domain dom1
apic1(config-vsan)# show run
# Command: show running-config vsan-domain dom1
# Time: Tue Jul 26 09:43:47 2016
  vsan-domain dom1
    vsan 1-10
    vlan 1-10
    fcoe vsan 2 vlan 2
    exit
apic1(config-vsan)# no fcoe vsan 2
apic1(config-vsan)# no vlan 1-10
apic1(config-vsan)# no vsan 1-10
apic1(config-vsan)# exit

################################################################################
NOTE: To remove a template-based VSAN to VLAN mapping use an alternate sequence:
################################################################################

apic1(config)# template vsan-attribute <template_name>
apic1(config-vsan-attr)# no fcoe vsan 2
```

Step 3 Delete the VSAN Domain.

The example deletes VSAN domain **dom1**.

Example:

```
apic1(config)# no vsan-domain dom1
```

Step 4 You can delete the associated tenant, EPG, and selectors if you do not need them.

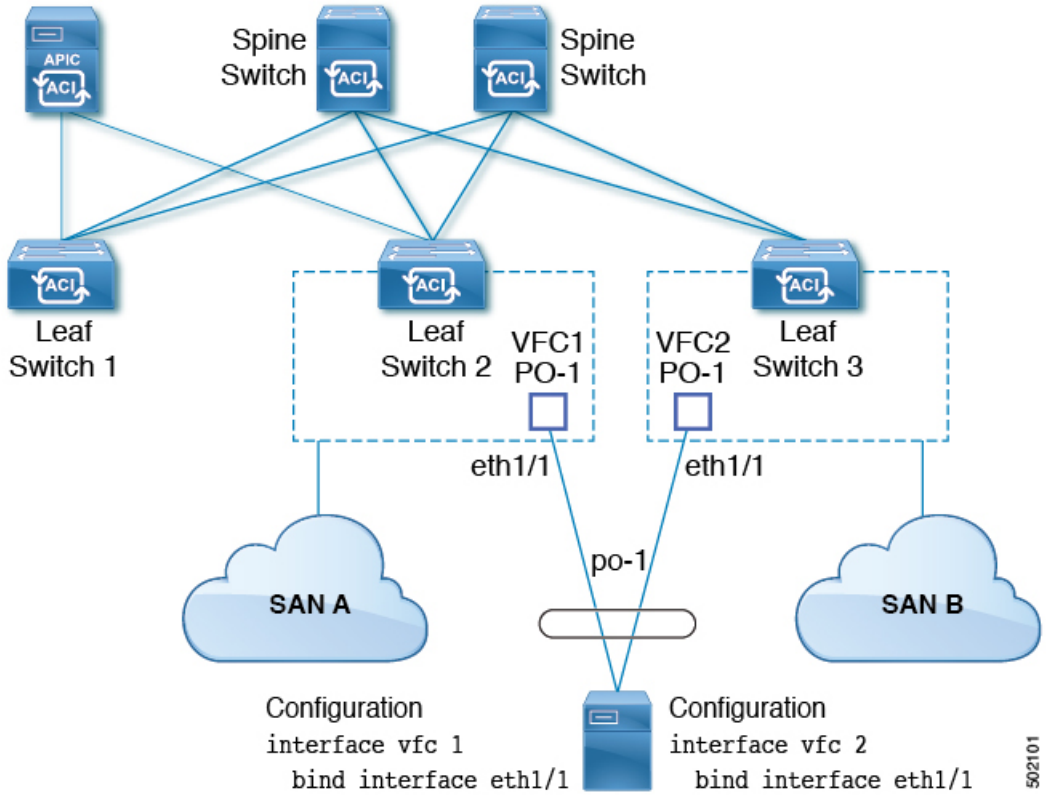
SAN Boot with vPC

Cisco ACI supports the SAN boot of initiators on Link Aggregation Control Protocol (LACP) based vPC. This limitation is specific to LACP-based port channels.

In the normal host-to-vPC topology, the host-facing vFC interface is bound to the vPC, and the vPC must be logically up before the vFC interface can come up. In this topology, a host will not be able to boot from SAN when LACP is configured on the vPC, because LACP on the host is typically implemented in the host driver and not in the adapter firmware.

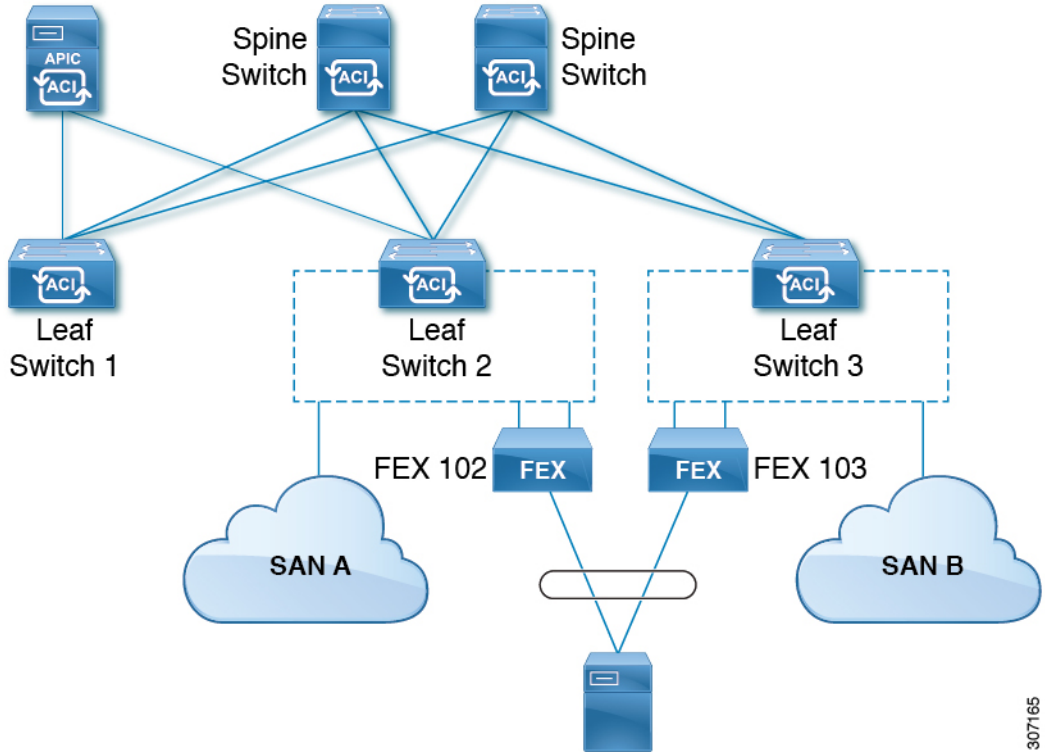
For SAN boot, the host-facing vFC interfaces are bound to port channel members instead of the port channel itself. This binding ensures that the host-side vFC comes up during a SAN boot as soon as the link on the CNA/Host Bus Adapter (HBA) comes up, without relying on the LACP-based port channel to form first.

Figure 30: SAN Boot Topology with vPC



Beginning with Cisco APIC Release 4.0(2), SAN boot is supported through a FEX host interface (HIF) port vPC, as shown in the following figure.

Figure 31: SAN Boot Topology with a FEX host interface (HIF) port vPC



Guidelines and Restrictions for SAN Boot with vPC

- Multi-member port channels are not supported.
- If a vFC is bound to a member port, the port channel cannot have more than 1 member.
- If a vFC is bound to a port channel, the port channel can have only one member port.

Configuring SAN Boot with vPC Using the GUI

To simplify the configuration, this procedure uses the **Configure Interface, PC, and vPC** wizard in **Fabric > Access Policies > Quickstart**.

Before you begin

This procedure assumes that the following items are already configured:

- VSAN Pool
- VLAN Pool
- VSAN Attributes, mapping VSANs in the VSAN pool to VLANs
- Fibre Channel domain (VSAN domain)
- Tenant, Application Profile

- Attached Entity Profile

Procedure

Step 1 On the APIC menu bar, navigate to **Fabric > Access Policies > Quickstart** and click *Configure an interface, PC, and VPC*.

Step 2 In the *Configure an interface, PC, and VPC* work area, in the **vPC Switch Pairs** toolbar, click **+** to create a switch pair. Perform the following actions:

a) From the **vPC Domain ID** text box, enter a number to designate the switch pair.
b) From the **Switch 1** drop-down list, select a leaf switch.

Only switches with interfaces in the same vPC policy group can be paired together.

c) From the **Switch 2** drop-down list, select a leaf switch.
d) click **Save** to save this switch pair.

Step 3 In the *Configure an interface, PC, and vPC* work area, click the large green **+** to select switches.
The **Select Switches To Configure Interfaces** work area opens with the **Quick** option selected by default.

Step 4 Select two switch IDs from the **Switches** drop-down list, and name the switch profile.

Step 5 Click the large green **+** again to configure the switch interfaces.

Step 6 In the **Interface Type** control, select **vPC**.

Step 7 For **Interfaces**, enter a single port number, such as `1/49`, that will be used on both switches as vPC members.

This action creates an interface selector policy. You can accept or change the name of the policy in the **Interface Selector Name** text box.

Step 8 In the **Interface Policy Group** control, select **Create One**.

Step 9 From the **Fibre Channel Interface Policy** text box, select **Create Fibre Channel Interface Policy** and perform the following actions.

a) In the **Name** field, type a name for the Fibre Channel interface policy.
b) From the **Port Mode** selector, select **F**.
c) From the **Trunk Mode** selector, select **trunk-on**.
d) Click **Submit**.

Step 10 From the **Port Channel Policy** text box, select **Create Port Channel Policy** and perform the following actions.

a) In the **Name** field, type a name for the port channel policy.
b) From the **Mode** drop-down list, select **LACP Active**.
c) From the **Control** selector, delete **Suspend Individual Port**.

Suspend Individual Port must be removed from the port channel; otherwise the physical interface will be suspended when LACP BPDU is not received from the host.

d) Click **Submit**.

Step 11 From the **Attached Device Type** drop-down list, select **Fibre Channel**.

Step 12 From the **Fibre Channel Domain** drop-down list, select your Fibre Channel domain (VSAN domain).

Step 13 Click **Save** to save this vPC configuration.

Step 14 Click **Save** to save this interface configuration.

Step 15 Click **Submit**.

Step 16 Expand **Tenants** > **Tenant** *name* > **Application Profiles** > *name* > **Application EPGs**.

Step 17 Right-click **Application EPGs**, select **Create Application EPG** and perform the following actions.

This EPG will be the Native EPG, in which the Native VLAN will be configured.

a) In the **Name** field, type a name for the EPG.
b) From the **Bridge Domain** drop-down list, select **Create Bridge Domain**.
c) In the **Name** field, type a name for the bridge domain.
d) From the **Type** control, select **regular**.
e) From the **VRF** drop-down list, choose the tenant VRF. If no VRF exists yet, select **Create VRF**, name the VRF and click **Submit**.
f) Click **Next**, **Next**, and **Finish** to return to **Create Application EPG**.
g) Click **Finish**.

Step 18 Expand the Native EPG created in the previous step.

Step 19 Right-click **Static Ports,** select **Deploy Static EPG On PC, VPC, or Interface** and perform the following actions.

a) From the **Path Type** control, select **Virtual Port Channel**.
b) From the **Path** drop-down list, select the port channel policy created for vPC.
c) From the **Port Encap** drop-down list, select **VLAN** and enter the number of an Ethernet VLAN.
d) From the **Deployment Immediacy** control, select **Immediate**.
e) From the **Mode** control, select **Access (802.1P)**.
f) Click **Submit**.

Step 20 Right-click **Application EPGs**, select **Create Application EPG** and perform the following actions.

This EPG will be the first of two EPGs, one for each SAN.

a) In the **Name** field, type a name for the EPG.
b) From the **Bridge Domain** drop-down list, select **Create Bridge Domain**.
c) In the **Name** field, type a name for the bridge domain.
d) From the **Type** control, select **fc**.
e) From the **VRF** drop-down list, choose the tenant VRF. If no VRF exists yet, select **Create VRF**, name the VRF and click **Submit**.
f) Click **Next**, **Next**, and **Finish** to return to **Create Application EPG**.
g) Click **Finish**.

Step 21 Repeat the previous step to create a second application EPG.

This second EPG will be used for the second SAN.

Step 22 Expand one of the two SAN EPGs, right-click **Fibre Channel (Paths)**, select **Deploy Fibre Channel** and perform the following actions.

a) From the **Path Type** control, select **Port**.
b) From the **Node** drop-down list, select one leaf of your switch pair.
c) From the **Path** drop-down list, select the Ethernet port number of your VPC.
d) In the **VSAN** text box, type the VSAN number prefixed by "vsan-".

For example, type "vsan-300" for VSAN number 300.

e) In the **VSAN Mode** control, select **Native**.
f) Click **Submit**.

Step 23 Expand the other of the two SAN EPGs and repeat the previous step, selecting the other leaf of your switch pair.

SAN Boot with vPC Configuration Using the CLI

This example assumes that the following items have been configured:

- A VLAN domain
- A tenant, application profile, and an application EPG
- A port channel template "Switch101-102_1-ports-49_PolGrp"

In this example, VSAN 200 is bound to physical Ethernet interface 1/49 on leaf 101 and VSAN 300 is bound to physical Ethernet interface 1/49 on leaf 102. The two interfaces are members of virtual port channel Switch101-102_1-ports-49_PolGrp.

```
apic1(config-leaf)# show running-config
# Command: show running-config leaf 101
# Time: Sat Sep  1 12:51:23 2018
  leaf 101

    interface ethernet 1/49
      # channel-group Switch101-102_1-ports-49_PolGrp vpc
      switchport trunk native vlan 5 tenant newtenant application AP1 epg epgNative
      port-direction downlink
      exit

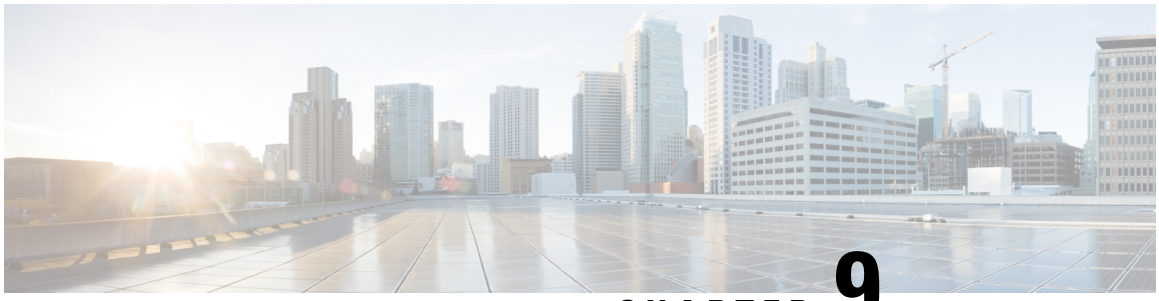
    # Port-Channel inherits configuration from "template port-channel
Switch101-102_1-ports-49_PolGrp"
    interface port-channel Switch101-102_1-ports-49_PolGrp
      exit

    interface vfc 1/49
     # Interface inherits configuration from "channel-group Switch101-102_1-ports-49_PolGrp"
 applied to interface ethernet 1/49
      switchport vsan 200 tenant newtenant application AP1 epg epg200
      exit

apic1(config-leaf)# show running-config
# Command: show running-config leaf 102
# Time: Sat Sep  1 13:28:02 2018
  leaf 102
    interface ethernet 1/49
      # channel-group Switch101-102_1-ports-49_PolGrp vpc
      switchport trunk native vlan 1 tenant newtenant application AP1 epg epgNative
      port-direction downlink
      exit

    # Port-Channel inherits configuration from "template port-channel
Switch101-102_1-ports-49_PolGrp"
    interface port-channel Switch101-102_1-ports-49_PolGrp
      exit

    interface vfc 1/49
     # Interface inherits configuration from "channel-group Switch101-102_1-ports-49_PolGrp"
 applied to interface ethernet 1/49
      switchport vsan 300 tenant newtenant application AP1 epg epg300
```



CHAPTER 9

Fibre Channel NPV

This chapter contains the following sections:

- Fibre Channel Connectivity Overview, on page 157
- NPV Traffic Management, on page 159
- SAN A/B Separation, on page 161
- SAN Port Channels, on page 162
- Fibre Channel N-Port Virtualization Guidelines and Limitations, on page 163
- Fibre Channel N-Port Virtualization Supported Hardware, on page 164
- Fibre Channel N-Port Virtualization Interoperability, on page 164
- Fibre Channel NPV GUI Configuration, on page 165
- Fibre Channel NPV NX-OS-Style CLI Configuration, on page 171
- Fibre Channel NPV REST API Configuration, on page 175

Fibre Channel Connectivity Overview

Cisco ACI supports Fibre Channel (FC) connectivity on a leaf switch using N-Port Virtualization (NPV) mode. NPV allows the switch to aggregate FC traffic from locally connected host ports (N ports) into a node proxy (NP port) uplink to a core switch.

A switch is in NPV mode after enabling NPV. NPV mode applies to an entire switch. Each end device connected to an NPV mode switch must log in as an N port to use this feature (loop-attached devices are not supported). All links from the edge switches (in NPV mode) to the NPV core switches are established as NP ports (not E ports), which are used for typical inter-switch links.

Note In the FC NPV application, the role of the ACI leaf switch is to provide a path for FC traffic between the locally connected SAN hosts and a locally connected core switch. The leaf switch does not perform local switching between SAN hosts, and the FC traffic is not forwarded to a spine switch.

FC NPV Benefits

FC NPV provides the following:

- Increases the number of hosts that connect to the fabric without adding domain IDs in the fabric. The domain ID of the NPV core switch is shared among multiple NPV switches.

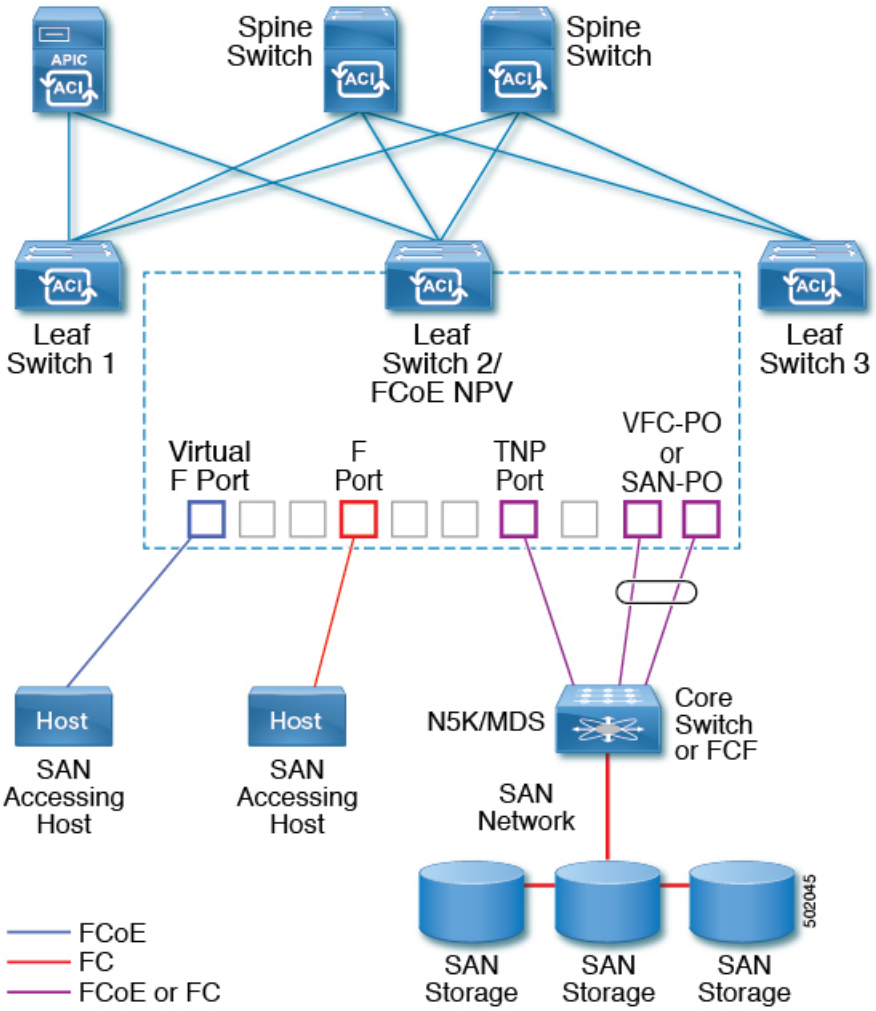
- FC and FCoE hosts connect to SAN fabrics using native FC interfaces.
- Automatic traffic mapping for load balancing. For newly added servers connected to NPV, traffic is automatically distributed among the external uplinks based on current traffic loads.
- Static traffic mapping. A server connected to NPV can be statically mapped to an external uplink.

FC NPV Mode

Feature-set fcoe-npv in ACI will be enabled automatically by default when the first FCoE/FC configuration is pushed.

FC Topology

The topology of various configurations supporting FC traffic over the ACI fabric is shown in the following figure:



- Server/storage host interfaces on the ACI leaf switch can be configured to function as either native FC ports or as virtual FC (FCoE) ports.

- An uplink interface to a FC core switch can be configured as any of the following port types:
  - native FC NP port
  - SAN-PO NP port
- An uplink interface to a FCF switch can be configured as any of the following port types:
  - virtual (vFC) NP port
  - vFC-PO NP port
- N-Port ID Virtualization (NPIV) is supported and enabled by default, allowing an N port to be assigned multiple N port IDs or Fibre Channel IDs (FCID) over a single link.
- Trunking can be enabled on an NP port to the core switch. Trunking allows a port to support more than one VSAN. When trunk mode is enabled on an NP port, it is referred to as a TNP port.
- Multiple FC NP ports can be combined as a SAN port channel (SAN-PO) to the core switch. Trunking is supported on a SAN port channel.
- FC F ports support 4/16/32 Gbps and auto speed configuration, but 8Gbps is not supported for host interfaces. The default speed is "auto."
- FC NP ports support 4/8/16/32 Gbps and auto speed configuration. The default speed is "auto."
- Multiple FDISC followed by Flogi (nested NPIV) is supported with FC/FCoE host and FC/FCoE NP links.
- An FCoE host behind a FEX is supported over an FCoE NP/uplink.
- Starting in the APIC 4.1(1) release, an FCoE host behind a FEX is supported over the Fibre Channel NP/uplink.
- All FCoE hosts behind one FEX can either be load balanced across multiple vFC and vFC-PO uplinks, or through a single Fibre Channel/SAN port channel uplink.
- SAN boot is supported on a FEX through an FCoE NP/uplink.
- Starting in the APIC 4.1(1) release, SAN boot is also supported over a FC/SAN-PO uplink.
- SAN boot is supported over vPC for FCoE hosts that are connected through FEX.

NPV Traffic Management

In most cases, Cisco recommends allowing all traffic to use all available uplinks. Use FC NPV traffic management only when automatic traffic engineering does not meet your network requirements.

Automatic Uplink Selection

NPV supports automatic selection of external (NP uplink) interfaces. When a server (host) interface is brought up, the external interface with the minimum load is selected from the available external interfaces in the same VSAN as the server interface.

When a new external interface becomes operational, the existing load is not redistributed automatically to include the newly available uplink. Server interfaces that become operational after the external interface can select the new uplink.

Traffic Maps

FC NPV supports traffic maps. A traffic map allows you to specify the external (NP uplink) interfaces that a server (host) interface can use to connect to the core switches.

Note When an FC NPV traffic map is configured for a server interface, the server interface must select only from the external interfaces in its traffic map. If none of the specified external interfaces are operational, the server remains in a non-operational state.

The FC NPV traffic map feature provides the following benefits:

- Facilitates traffic engineering by allowing configuration of a fixed set of external interfaces for a specific server interface (or range of server interfaces).
- Ensures correct operation of the persistent FC ID feature; this is because a server interface will always connect to the same external interface (or one of a specified set of external interfaces) by providing the same traffic path after an interface reinitialization or switch reboot.

Disruptive Auto Load Balancing of Server Logins across NP Links

FC NPV supports disruptive load balancing of server logins. When disruptive load balancing is enabled, FC NPV redistributes the server interfaces across all available NP uplinks when a new NP uplink becomes operational. To move a server interface from one NP uplink to another NP uplink, FC NPV forces reinitialization of the server interface so that the server performs a new login to the core switch.

Only server interfaces that are moved to a different uplink are reinitialized. A system message is generated for each server interface that is moved.

Note Redistributing a server interface causes traffic disruption to the attached end devices. Adding a member to the existing port-channel does not trigger disruptive auto load-balance.

To avoid disruption of server traffic, you should enable this feature only after adding a new NP uplink, and then disable it again after the server interfaces have been redistributed.

If disruptive load balancing is not enabled, you can manually reinitialize some or all of the server interfaces to distribute server traffic to new NP uplink interfaces.

FC NPV Traffic Management Guidelines

When deploying FC NPV traffic management, follow these guidelines:

- Use FC NPV traffic management only when automatic traffic engineering does not meet your network requirements.

- You do not need to configure traffic maps for all server interfaces. By default, FC NPV will use automatic traffic management.
- Server interfaces configured to use a set of NP uplink interfaces cannot use any other available NP uplink interfaces, even if none of the configured interfaces are available.
- When disruptive load balancing is enabled, a server interface may be moved from one NP uplink to another NP uplink. Moving between NP uplink interfaces requires FC NPV to relogin to the core switch, causing traffic disruption.
- To link a set of servers to a specific core switch, associate the server interfaces with a set of NP uplink interfaces that all connect to that core switch.
- Configure Persistent FC IDs on the core switch and use the traffic map feature to direct server interface traffic onto NP uplinks that all connect to the associated core switch.
- When initially configuring traffic map pinning, you must shut the server host port before configuring the first traffic map.
- If traffic mapping is configured for more than one uplink, when removing the traffic map through which a host has logged in, you must first shut the host before removing the traffic map.
- While configuring a traffic map for an FCoE host behind a FEX, you can map one host to either multiple FCoE NP/uplinks (VFC or VFC-PO) or to a single Fibre Channel/SAN port channel NP/uplink.

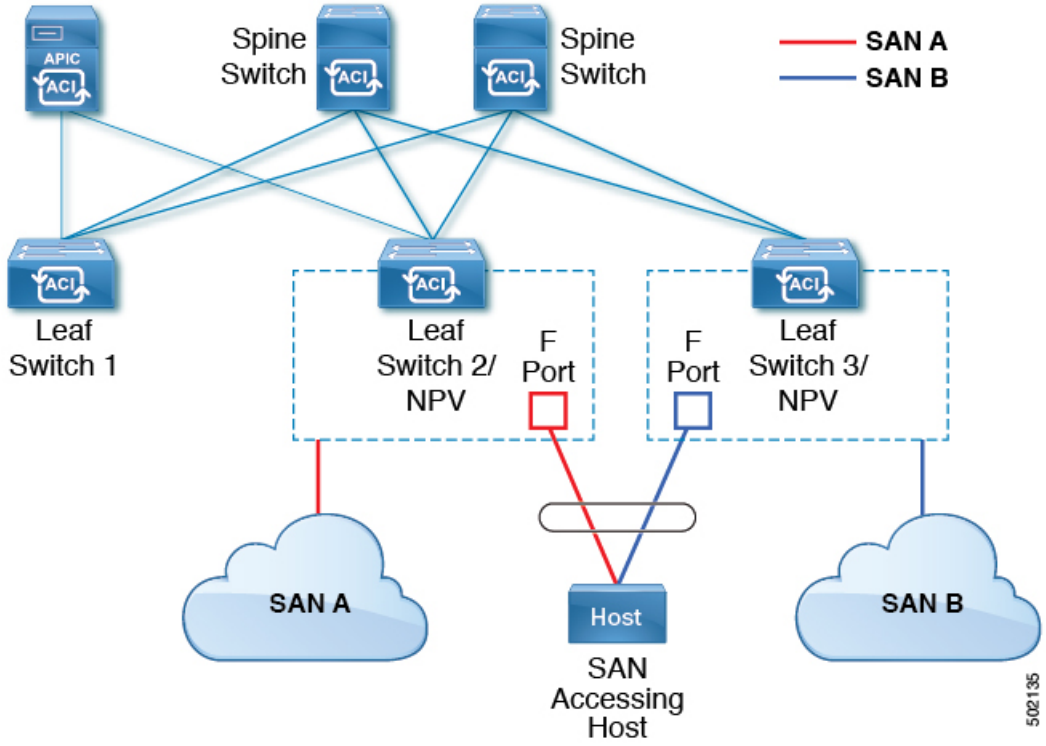


Note When a server is statically mapped to an external interface, the server traffic is not redistributed in the event that the external interface becomes down for any reason.

SAN A/B Separation

SAN A and SAN B separation ensures that SAN connectivity is available even if one of the fabric components fails. SAN A and SAN B separation can be achieved physically or logically by separating the VSANs that are carried across the fabric.

Figure 32: SAN A/B Separation



SAN Port Channels

About SAN Port Channels

- A SAN port channel is a logical interface that combines a set of FC interfaces connected to the same Fibre Channel node and operates as one link.
- SAN port channels support bandwidth utilization and availability.
- SAN port channels on Cisco ACI switches are used to connect to FC core switches and to provide optimal bandwidth utilization and transparent failover between the uplinks of a VSAN.

SAN Port Channel Guidelines and Limitations

- The maximum number of active port channels (SAN port channels plus VFC uplink/NP port channels) on the Cisco ACI switch is seven. Any additional configured port channels remain in the **errdisabled** state until you shut or delete one of the existing active port channels. After you shut/delete an existing active port channel, shut/no shut the **errdisabled** port channel to bring it up.
- The maximum number of FC interfaces that can be combined into a SAN port channel is limited to 16.
- The default channel mode on Cisco ACI switches for SAN port channels is **active**; this cannot be changed.
- When a SAN port channel is connected to a Cisco FC core switch, only channel mode active is supported. Channel mode active must be configured on the Cisco FC core switch.

About SAN Port Channel Modes

A SAN port channel is configured with channel mode active by default. When active, the member ports initiate port channel protocol negotiation with the peer port regardless of the channel-group mode of the peer port. If the peer port, while configured in a channel group, does not support the port-channel protocol, or responds with a nonnegotiable status, the port channel is disabled. The active port channel mode allows automatic recovery without explicitly enabling and disabling the port-channel-member ports at either end.

Fibre Channel N-Port Virtualization Guidelines and Limitations

When configuring Fibre Channel N-Port Virtualization (NPV), note the following guidelines and limitations:

- Fibre Channel NP ports support trunk mode, but Fibre Channel F ports do not.
- On a trunk Fibre Channel port, internal login happens on the highest VSAN.
- On the core switch, the following features must be enabled:

```
feature npiv
feature fport-channel-trunk
```

- To use an 8G uplink speed, you must configure the IDLE fill pattern on the core switch.



Note Following is an example of configuring IDLE fill pattern on a Cisco MDS switch:

```
Switch(config)# int fc2/3
Switch(config)# switchport fill-pattern IDLE speed 8000
Switch(config)# show run int fc2/3

interface fc2/3
switchport speed 8000
switchport mode NP
switchport fill-pattern IDLE speed 8000
no shutdown
```

- Fibre Channel NPV support is limited to the Cisco N9K-C93180YC-FX switch.
- You can use ports 1 through 48 for Fibre Channel configuration. Ports 49 through 54 cannot be Fibre Channel ports.
- If you convert a port from Ethernet to Fibre Channel or the other way around, you must reload the switch. Currently, you can convert only one contiguous range of ports to Fibre Channel ports, and this range must be a multiple of 4, ending with a port number that is a multiple of 4. For example, 1-4, 1-8, or 21-24.
- Fibre Channel Uplink (NP) connectivity to Brocade Port Blade Fibre Channel 16-32 is not supported when a Cisco N9K-93180YC-FX leaf switch port is configured in 8G speed.
- The selected port speed must be supported by the SFP. For example, because a 32G SFP supports 8/16/32G, a 4G port speed requires an 8G or 16G SFP. Because a 16G SFP supports 4/8/16G, a 32G port speed requires a 32G SFP.
- Speed autonegotiation is supported. The default speed is 'auto'.
- You cannot use Fibre Channel on 40G and breakout ports.

- FEX cannot be directly connected to FC ports.
- FEX HIF ports cannot be converted to FC.
- Reloading a switch after changing a switch's port profile configuration interrupts traffic through the data plane.

Fibre Channel N-Port Virtualization Supported Hardware

Fibre Channel N-Port Virtualization (FC NPV) is supported on the following switches:

- N9K-C93108TC-FX
- N9K-C93180YC-FX

The following Fibre Channel small form-factor pluggable (SFP) transceivers are supported:

- DS-SFP-FC8G-SW: 2/4/8G (2G is not a supported FC NPV port speed)
- DS-SFP-FC16G-SW: 4/8/16G (not compatible when FC NPV port speed is 32G)
- DS-SFP-FC32G-SW: 8/16/32G (not compatible when FC NPV port speed is 4G)

The supported NPIV core switches are the Cisco Nexus 5000 series, Nexus 6000 series, Nexus 7000 series (FCoE), and Cisco MDS 9000 series multilayer switches.

Fibre Channel N-Port Virtualization Interoperability

The following table lists third party products with which the Fibre Channel N-port virtualization (FC NPV) feature of Cisco Application Policy Infrastructure Controller (APIC) was tested for interoperability.

Table 6: Third Party Products That Are Supported With FC NPV

Third Party Switch Vendor	Brocade
Third Party Hardware Model	DS-6620B
Third Party Software Release	8.2.1a
Cisco NX-OS Release	14.1(1) and later
Cisco Nexus 9000 Model	N9K-C93180YC-FX
Interoperability Mode	NA (NPV)
Cisco SFP Module	DS-SFP-FC32G-SW
Third Party SFP Module	Brocade-32G

Fibre Channel NPV GUI Configuration

Configuring a Native Fibre Channel Port Profile Using the GUI

This procedure configures a set of native Fibre Channel (FC) F ports for connecting to Fibre Channel hosts, such as servers.

To simplify the configuration, this procedure uses the **Configure an Interface, PC, and vPC** wizard.

Procedure

Step 1 On the APIC menu bar, navigate to **Fabric > Access Policies > Quickstart** and click *Configure an interface, PC, and vPC*.

Step 2 In the **Configured Switch Interfaces** toolbar, click **+** to create a switch profile. Perform the following actions:

This switch profile configures your server host ports. Another switch profile configures your uplink ports.

a) From the **Switches** drop-down list, choose your NPV leaf switch.

This action automatically creates a leaf switch profile. You can accept or change the name of the leaf switch profile in the **Switch Profile Name** text box.

b) Click the large green **+** on the ports drawing to open more interface settings.
c) For **Interface Type**, select **FC** to specify Fibre Channel host interface ports (F ports).
d) For **Interfaces**, enter a port range for the FC ports.

Only one contiguous range of ports can be converted to FC ports. This range must be a multiple of 4 ending with a port number that is a multiple of 4 (for example, 1-4, 1-8, and 21-32 are valid ranges).

This action creates an interface selector policy. You can accept or change the name of the policy in the **Interface Selector Name** text box.

Note
Port conversion from Ethernet to FC requires a reload of the switch. After the interface policy is applied, a notification alarm appears in the GUI, prompting you to reload the switch. During a switch reload, communication to the switch is interrupted, resulting in timeouts when trying to access the switch.

e) From the **Policy Group Name** drop-down list, select **Create FC Interface Policy Group**.
f) In the **Create FC Interface Policy Group** dialog box, type a name in the **Name** field.
g) In the **Fibre Channel Interface Policy** drop-down list, select **Create Fibre Channel Interface Policy**.
h) In the **Create Fibre Channel Interface Policy** dialog box, type a name in the **Name** field and configure the following settings:

Field	Setting
Port Mode	For host interfaces, select **F**.
Trunk Mode	For host interfaces, select **trunk-off**.
Speed	Select **auto** (default).

Field	Setting
Auto Max Speed	Auto Max Speed configuration is applicable only when speed is **auto**. Auto Max Speed is to limit maximum speed when speed is in auto mode.
Receive Buffer Credit	Select **64**.

i) Click **Submit** to save the Fibre Channel interface policy and return to the **Create FC Interface Policy Group** dialog box.
j) From the **Attached Entity Profile** drop-down list, choose **Create Attachable Access Entity Profile**.

The attachable entity profile option specifies the interfaces where the leaf access port policy is deployed.

k) In the **Name** field, enter a name for the attachable entity policy.
l) In the **Domains (VMM, Physical, or External) To Be Associated To Interfaces** toolbar, click + to add a domain profile.
m) From the **Domain Profile** drop-down list, choose **Create Fibre Channel Domain**.
n) In the **Name** field, enter a name for the Fibre Channel domain.
o) From the **VSAN Pool** drop-down list, choose **Create VSAN Pool**.
p) In the **Name** field, enter a name for the VSAN pool.
q) In the **Encap Blocks** toolbar, click + to add a VSAN range.
r) In the **Create VSAN Ranges** dialog box, enter **From** and **To** VSAN numbers.
s) For **Allocation Mode**, select **Static Allocation** and click **OK**.
t) In the **Create VSAN Ranges** dialog box, click **Submit**.
u) In the **Create Fibre Channel Domain** dialog box, click **Submit**.

Note
In the Fibre Channel Domain, when using native FC ports instead of FCoE, it is not necessary to configure a VLAN pool or VSAN attributes.

v) In the **Create Attachable Access Entity Profile** dialog box, click **Update** to select the Fibre Channel domain profile and click **Submit**.
w) In the **Create FC Policy Group** dialog box, click **Submit**.
x) In the **Configure Interface, PC, and vPC** dialog box, click **Save** to save this switch profile for your server host ports.

Note
Port conversion from Ethernet to FC requires a reload of the switch. After the interface policy is applied, a notification alarm appears in the GUI, prompting you to reload the switch. During a switch reload, communication to the switch is interrupted, resulting in timeouts when trying to access the switch.

Note
When you change a port profile on a switch, for example reconfigure an uplink as downlink and reload the switch, communication to the switch is interrupted until the switch gets its configuration from Cisco APIC.

In **Fabric > Access Policies > Switches > Leaf Switches > Profiles >** ***name***, the Fibre Channel port profile appears in the **Associated Interface Selector Profiles** list in the **Leaf Profiles** work pane.

What to do next

- Configure a Fibre Channel uplink connection profile.
- Deploy the server ports and uplink ports in a tenant to connect to a Fibre Channel core switch.

Configuring a Native FC Port Channel Profile Using the GUI

This procedure configures a native Fibre Channel port channel (FC PC) profile for an uplink connection to a Fibre Channel core switch.



Note This procedure can also be performed using the **Configure Interface, PC, and vPC** wizard.

Before you begin

Configure your uplink connections, including an attachable entity profile.

Procedure

Step 1 Expand **Fabric > Access Policies > Interfaces > Leaf Interfaces > Profiles**.

Step 2 Right click **Profiles** and click **Create Leaf Interface Profile**.

Step 3 In the **Create Leaf Interface Profile** dialog box, perform the following steps:

a) In the **Name** field, enter a name for the leaf interface profile.

b) In the **Interface Selectors** toolbar, click **+** to open the **Create Access Port Selector** dialog box.

c) In the **Name** field, enter a name for the port selector.

d) In the **Interface IDs** field, enter a port range for the FC PC ports.

The port channel can have a maximum of 16 ports.

Only one contiguous range of ports can be converted to FC ports. This range must be a multiple of 4 ending with a port number that is a multiple of 4 (for example, 1-4, 1-8, and 21-32 are valid ranges).

Note
Port conversion from Ethernet to FC requires a reload of the switch. After the interface policy is applied, a notification alarm appears in the GUI, prompting you to reload the switch manually. During a switch reload, communication to the switch is interrupted, resulting in timeouts when trying to access the switch.

e) From the **Interface Policy Group** drop-down list, choose **Create FC PC Interface Policy Group**.

f) In the **Name** field, enter a name for the FC PC interface policy group.

g) From the **Fibre Channel Interface Policy** drop-down list, choose **Create Fibre Channel Interface Policy**.

h) In the **Name** field, enter a name for the FC PC interface policy.

i) In the **Create Interface FC Policy** dialog box, type a name in the **Name** field and configure the following settings:

Field	Setting
Port Mode	For uplink interfaces, select **NP**.

Field	Setting
Trunk Mode	For uplink interfaces, select **trunk-on**.

j) Click **Submit** to save the FC PC interface policy and return to the **Create FC PC Interface Policy Group** dialog box.
k) From the **Port Channel Policy** drop-down list, choose **Create Port Channel Policy**.
l) In the **Name** field, enter a name for the port channel policy.

The other settings in this menu can be ignored.

m) Click **Submit** to save the port channel policy and return to the **Create FC PC Interface Policy Group** dialog box.
n) From the **Attached Entity Profile** drop-down list, choose the existing attachable entity profile.
o) Click **Submit** to return to the **Create Access Port Selector** dialog box.
p) Click **OK** to return to the **Create Leaf Interface Profile** dialog box.
q) Click **OK** to return to the **Leaf Interfaces - Profiles** work pane.

Step 4 Expand **Fabric > Access Policies > Switches > Leaf Switches > Profiles**.

Step 5 Right click the leaf switch profile that you created and click **Create Interface Profile**.

Step 6 In the **Create Interface Profile** dialog box, perform the following steps:

a) From the **Interface Select Profile** drop-down list, choose the leaf interface profile that you created for the port channel.
b) Click **Submit** to return to the **Leaf Interfaces - Profiles** work pane.

Note
Port conversion from Ethernet to FC requires a reload of the switch. After the interface policy is applied, a notification alarm appears in the GUI, prompting you to reload the switch. During a switch reload, communication to the switch is interrupted, resulting in timeouts when trying to access the switch.

In **Fabric > Access Policies > Switches > Leaf Switches > Profiles > *name***, the FC port channel profile appears in the **Associated Interface Selector Profiles** list in the work pane.

What to do next

Deploy the server ports and uplink ports in a tenant to connect to a Fibre Channel core switch.

Deploying Fibre Channel Ports

This procedure activates the Fibre Channel server host ports and uplink ports.

Before you begin

- Configure Fibre Channel (FC) server host port profiles (F ports).
- Configure FC uplink port profiles (NP or TNP ports).
- Configure a leaf switch profile that includes two associated interface selector profiles — one for host ports and one for uplink ports.

Procedure

Step 1 Expand **Tenants > Tenant *name* > Application Profiles**

If the tenant does not exist, you must create a tenant.

Step 2 Right click **Application Profiles**, click **Create Application Profile**, and perform the following actions:

a) In the **Name** field, enter a name for the application profile.
b) Click **Submit**.

Step 3 Expand **Tenants > Tenant *name* > Application Profiles > *name*** > **Application EPGs**

Step 4 Right click **Application EPGs** and click **Create Application EPG**, and perform the following actions:

Step 5 In the **Create Application EPG** dialog box, perform the following actions:

a) In the **Name** field, enter a name for the application EPG.
b) Configure the following settings:

Field	Setting
Intra EPG Isolation	Select **Unenforced**.
Preferred Group Member	Select **Exclude**.
Flood on Encapsulation	Select **Disabled**.

c) From the **Bridge Domain** drop-down list, select **Create Bridge Domain**.
d) In the **Name** field, enter a name for the bridge domain.
e) For **Type**, select **fc** to specify a Fibre Channel bridge domain.
f) From the **VRF** drop-down list, select **Create VRF**.
g) In the **Name** field, enter a name for the VRF.
h) Click **Submit** to return to the **Create Bridge Domain** dialog box.
i) Click **Next**, then **Next**, then **Finish** to return to the **Create Application EPG** dialog box.
j) Click **Finish**.

Step 6 Expand **Tenants > Tenant *name* > Application Profiles > *name* > Application EPGs > *name* > Domains (VMs and Bare-Metals)**.

Step 7 Right click **Domains (VMs and Bare-Metals)** and click **Add Fibre Channel Domain Association**, and perform the following actions:

a) From the **Fibre Channel Domain Profile** drop-down list, select the Fibre Channel domain that you created when you configured your host ports.
b) Click **Submit**.

Step 8 Expand **Tenants > Tenant *name* > Application Profiles > *name* > Application EPGs > *name* > Fibre Channel (Paths)** and perform the following actions:

This step deploys the server host ports.

a) Right click **Fibre Channel (Paths)** and click **Deploy Fibre Channel**.
b) In the **Path Type** control, click **Port**.
c) From the **Node** drop-down list, choose the leaf switch.
d) From the **Path** drop-down list, choose the leaf switch port that is configured as a server host port.

e) In the **VSAN** field, enter the port VSAN.
f) In the **VSAN Mode** control, click **Native**.
g) Verify that the **Type** is fcoe.
h) (Optional) If you require a traffic map, use the **Pinning Label** drop-down list.

Note
If multiple uplink ports are available and you want this host port to always direct its FLOGI to a specific uplink, you can create a pinning profile (traffic map) to associate the host port to the uplink port. Otherwise, hosts are load-balanced among the available uplink ports.

i) Click **Submit**.
j) Repeat from **Step a** for each Fibre Channel host port.

Step 9 Expand **Tenants > Tenant *name* > Application Profiles > *name* > Application EPGs > *name* > Fibre Channel (Paths)** and perform the following actions:

This step deploys the uplink port channel.

a) Right click **Fibre Channel (Paths)** and click **Deploy Fibre Channel**.
b) In the **Path Type** control, click **Direct Port Channel**.
c) From the **Path** drop-down list, choose the uplink port channel.
d) In the **VSAN** field, enter the port default VSAN.
e) In the **VSAN Mode** control, click **Native** for a port VSAN or **Regular** for a trunk VSAN.
f) Verify that the **Type** is fcoe.
g) Click **Submit**.
h) Repeat from **Step a** for each Fibre Channel uplink port or port channel.

Configuring a Traffic Map for a Fibre Channel Port

In an application in which multiple uplink ports are available, server traffic by default is load-balanced among the available uplink ports. In some cases, it might be necessary to have a server send its login request (FLOGI) to one or more specific uplink ports or port channels. In such cases, you can create a pinning profile (traffic map) to associate the server port to those uplink ports or port channels.

This procedure assumes that you have already configured one or more server ports and one or more uplink ports or port channels. Because the server ports have already been configured, you must first shut (disable) any server port that is to be mapped to an uplink. After configuring the traffic map, re-enable the port.

Before you begin

This procedure assumes that the following items are already configured:

- Server ports (F ports) and uplink ports or port channels (NP ports)
- A tenant, including an application profile and application EPG

Note Before creating a pinning profile (traffic map), you must shut the server port that is to be mapped to an uplink.

Procedure

Step 1 In the **Fabric** > **Inventory** > **Pod** ***n*** > **Leaf** ***n*** > **Interfaces** > **FC Interfaces** work pane, select and disable the server interface port that is to be mapped to an uplink.

Step 2 Expand **Tenants** > **Tenant** ***name*** > **Application Profiles** > ***application profile name*** > **Application EPGs** > ***EPG name*** > **Fibre Channel (Paths)** and perform the following actions:

a) Right click **Fibre Channel (Paths)** and click **Deploy Fibre Channel**.
b) In the **Path Type** control, click **Port**.
c) From the **Node** drop-down list, choose the leaf switch.
d) From the **Path** drop-down list, choose the server port that is to be mapped to a specific uplink port.
e) In the **VSAN** field, enter the port default VSAN.
f) In the **VSAN Mode** control, click **Native**.
g) Verify that the **Type** is fcoe.
h) From the **Pinning Label** drop-down list, choose **Create Pinning Profile**.
i) In the **Name** field, enter a name for the traffic map.
j) In the **Path Type** control, click **Port** to connect to a single NP uplink port or **Direct Port Channel** to connect to an FC port channel.

   If you choose **`Port`** for the path type, you must also choose the leaf switch from the **Node** drop-down list that appears.

   If you choose **`Direct Port Channel`** for the path type, you must also choose the FC PC you have defined in Interface Policy Group.

k) From the **Path** drop-down list, choose the uplink port or port channel to which the server port will be mapped.
l) Click **Submit** to return to the **Deploy Fibre Channel** dialog box.
m) Click **Submit**.

Step 3 In the **Fabric** > **Inventory** > **Pod** ***n*** > **Leaf** ***n*** > **Interfaces** > **FC Interfaces** work pane, select and re-enable the server interface port that is mapped to an uplink.

Fibre Channel NPV NX-OS-Style CLI Configuration

Configuring Fibre Channel Interfaces Using the CLI

On an NPV-enabled leaf switch, you can convert universal ports to Fibre Channel (FC) ports. The FC ports can be either F ports or NP ports, and NP ports can form a port channel.

Procedure

Step 1 Convert a range of ports from Ethernet to Fibre Channel.

Example:

```
apic1(config)# leaf 101
```

```
apic1(config-leaf)# slot 1
apic1(config-leaf-slot)# port 1 12 type fc
```

This example converts ports 1/1-12 on leaf 101 to Fibre Channel ports. The **[no]** form of the **port type fc** command converts the ports from Fibre Channel back to Ethernet.

Note
The conversion of ports takes place only after a reboot of the leaf switch.

Currently only one contiguous range of ports can be converted to FC ports, and this range must be a multiple of 4 ending with a port number that is a multiple of 4 (for example, 1-4, 1-8, or 21-24).

Step 2 Configure all Fibre channel interfaces.

Example:

```
apic1(config)# leaf 101
apic1(config-leaf)# interface fc 1/1
apic1(config-leaf-fc-if)# switchport mode [f | np]
apic1(config-leaf-fc-if)# switchport rxbbcredit <16-64>
apic1(config-leaf-fc-if)# switchport speed [16G | 32G | 4G | 8G | auto | unknown]
apic1(config-leaf-fc-if)# switchport trunk-mode [ auto | trunk-off | trunk-on | un-init]
apic1(config-leaf-fc-if)# switchport [trunk allowed] vsan <1-4093> tenant <name>  \
                              application <name> epg <name>
```

Note
FC host interfaces (F ports) do not support a speed configuration of 8Gbps.

A FC interface can be configured in access mode or trunk mode. To configure the FC port in access mode, use the following command format:

Example:

```
apic1(config-leaf-fc-if)# switchport vsan 2 tenant t1 application a1 epg e1
```

To configure a FC port in trunk mode, use the following command format:

Example:

```
apic1(config-leaf-fc-if)# switchport trunk allowed vsan 4 tenant t1 application a1 epg e1
```

To configure a FC port channel, configure a FC port interface template and apply it to FC interfaces that will be members of the FC port-channel.

The port channel can have a maximum of 16 members.

Example:

```
apic1(config)# template fc-port-channel my-fc-pc
apic1(config-fc-po-ch-if)# lacp max-links 4
apic1(config-fc-po-ch-if)# lacp min-links 1
apic1(config-fc-po-ch-if)# vsan-domain member dom1
apic1(config-fc-po-ch-if)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface fc 1/1-2
apic1(config-leaf-fc-if)# fc-channel-group my-fc-pc
apic1(config-leaf-fc-if)# exit
apic1(config-leaf)# interface fc-port-channel my-fc-pc
apic1(config-leaf-fc-pc)# switchport mode [f | np]
```

```
apic1(config-leaf-fc-pc)# switchport rxbbcredit <16-64>
apic1(config-leaf-fc-pc)# switchport speed [16G | 32G | 4G | 8G | auto | unknown]
apic1(config-leaf-fc-pc)# switchport trunkmode [ auto | trunk-off | trunk-on | un-init]
```

Configuring Fibre Channel NPV Policies Using the CLI

Before you begin

Leaf switch ports to be used in an NPV application have been converted to Fibre Channel (FC) ports.

Procedure

Step 1 Create a template of a Fibre Channel F port policy group.

Example:

```
apic1(config)# template fc-policy-group my-fc-policy-group-f-ports
apic1(config-fc-pol-grp-if)# vsan-domain member dom1
apic1(config-fc-pol-grp-if)# switchport mode f
apic1(config-fc-pol-grp-if)# switchport trunk-mode trunk-off
```

You can configure other switchport settings, such as speed.

Step 2 Create a template of a Fibre Channel NP port policy group.

Example:

```
apic1(config)# template fc-policy-group my-fc-policy-group-np-ports
apic1(config-fc-pol-grp-if)# vsan-domain member dom1
apic1(config-fc-pol-grp-if)# switchport mode np
apic1(config-fc-pol-grp-if)# switchport trunk-mode trunk-on
```

You can configure other switchport settings, such as speed.

Step 3 Create a fabric-wide Fibre Channel policy.

Example:

```
apic1(config)# template fc-fabric-policy my-fabric-fc-policy
apic1(config-fc-fabric-policy)# fctimer e-d-tov 1000
apic1(config-fc-fabric-policy)# fctimer r-a-tov 5000
apic1(config-fc-fabric-policy)# fcoe fcmap 0E:FC:01
```

Step 4 Create a Fibre Channel port channel policy.

Example:

```
apic1(config)# template fc-port-channel my-fc-pc
apic1(config-fc-po-ch-if)# lacp max-links 4
apic1(config-fc-po-ch-if)# lacp min-links 1
```

```
apic1(config-fc-po-ch-if)# vsan-domain member dom1
```

Step 5 Create a leaf-wide Fibre Channel policy group.

Example:

```
apic1(config)# template fc-leaf-policy my-fc-leaf-policy
apic1(config-fc-leaf-policy)# npv auto-load-balance disruptive
apic1(config-fc-leaf-policy)# fcoe fka-adv-period 10
```

Note
The policy commands that are shown here are only examples, and are not mandatory settings.

Step 6 Create a leaf policy group.

```
apic1(config)# template leaf-policy-group lpg1
apic1(config-leaf-policy-group)# inherit fc-fabric-policy my-fabric-fc-policy
apic1(config-leaf-policy-group)# inherit fc-leaf-policy my-fc-leaf-policy
```

The leaf policy group is created by inheriting FC-related policies.

Step 7 Create a leaf profile to apply a leaf-policy-group to a leaf-group.

Example:

```
apic1(config)# leaf-profile my-leaf-profile
apic1(config-leaf-profile)# leaf-group my-leaf-group
apic1(config-leaf-group)# leaf 101
apic1(config-leaf-group)# leaf-policy-group lpg1
```

This example applies fabric-wide FC policies and leaf-wide FC policies that are grouped into a leaf policy group lpg1 to leaf 101.

Step 8 Create a leaf interface profile and apply a fc-policy-group to a set of FC interfaces.

Example:

```
apic1(config)# leaf-interface-profile my-leaf-interface-profile
apic1(config-leaf-if-profile)# leaf-interface-group my-leaf-interface-group
apic1(config-leaf-if-group)# fc-policy-group my-fc-policy-group-f-ports
apic1(config-leaf-if-group)# interface fc 1/1-10
```

Configuring an NPV Traffic Map Using the CLI

This procedure maps traffic coming from a FC/FCoE server (host) interface to a FC/FCoE external (uplink) interface configured in NP mode.

Before you begin

All server interfaces must be F ports and all uplink interfaces must be NP ports.

Procedure

Example:

```
apic1(config)# leaf 101
apic1(config-leaf)# npv traffic-map server-interface  \
    { vfc <slot/port> | vfc-po <po-name> |fc <slot/port> } \
    label <name> tenant <tn> app <ap> epg <ep>
apic1(config-leaf)# npv traffic-map external-interface \
    { vfc <slot/port> | vfc-po <po-name> |fc <slot/port> } \
    tenant <tn> label <name>
```

Example:

```
apic1(config)# leaf 101
apic1(config-leaf)# npv traffic-map server-interface vfc 1/1 label serv1 tenant t1 app ap1 epg epg1
apic1(config-leaf)# npv traffic-map external-interface vfc-po my-fc-pc tenant t1 label ext1
```

Fibre Channel NPV REST API Configuration

Configuring FC Connectivity Using the REST API

You can configure FC-enabled interfaces and EPGs accessing those interfaces using the FC protocol with the REST API.

Procedure

Step 1 To create a VSAN pool, send a post with XML such as the following example. The example creates VSAN pool myVsanPool1 and specifies the range of VSANs to be included as vsan-50 to vsan-60:

Example:

```
https://apic-ip-address/api/mo/uni/infra/vsanns-[myVsanPool1]-static.xml

<fvnsVsanInstP allocMode="static" name="myVsanPool1">
  <fvnsVsanEncapBlk from="vsan-50" name="encap" to="vsan-60"/>
</fvnsVsanInstP>
```

Step 2 To create a Fibre Channel domain, send a post with XML such as the following example. The example creates Fibre Channel domain (VSAN domain) myFcDomain1 and associates it with the VSAN pool myVsanPool1:

Example:

```
https://apic-ip-address/api/mo/uni/fc-myFcDomain1.xml

<fcDomP name="myFcDomain1">
  <fcRsVsanNs tDn="uni/infra/vsanns-[myVsanPool1]-static"/>
```

```
</fcDomP>
```

Step 3 To create an Attached Entity Policy (AEP) for the FC ports, send a post with XML such as the following example. The example creates the AEP myFcAEP1 and associates it with the Fibre Channel domain myFcDomain1:

Example:

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
<infraInfra>
    <infraAttEntityP name="myFcAEP1">
        <infraRsDomP tDn="uni/fc-myFcDomain1"/>
    </infraAttEntityP>
</infraInfra>
</polUni>
```

Step 4 To create a FC interface policy and a policy group for server host ports, send a post with XML. This example executes the following requests:

- Creates a FC interface policy myFcHostIfPolicy1 for server host ports. These are F ports with no trunking.
- Creates a FC interface policy group myFcHostPortGroup1 that includes the FC host interface policy myFcHostIfPolicy1.
- Associates the policy group to the FC interface policy to convert these ports to FC ports.
- Creates a host port profile myFcHostPortProfile.
- Creates a port selector myFcHostSelector that specifies ports in range 1/1-8.
- Creates a node selector myFcNode1 that specifies leaf node 104.
- Creates a node selector myLeafSelector that specifies leaf node 104.
- Associates the host ports to the leaf node.

Example:

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
    <infraInfra>
        <fcIfPol name="myFcHostIfPolicy1" portMode="f" trunkMode="trunk-off" speed="auto"/>
        <infraFuncP>
            <infraFcAccPortGrp name="myFcHostPortGroup1">
                <infraRsFcL2IfPol tnFcIfPolName="myFcHostIfPolicy1" />
            </infraFcAccPortGrp>
        </infraFuncP>
        <infraAccPortP name="myFcHostPortProfile">
            <infraHPortS name="myFcHostSelector" type="range">
                <infraPortBlk name="myHostPorts" fromCard="1" toCard="1" fromPort="1" toPort="8" />
                <infraRsAccBaseGrp tDn="uni/infra/funcprof/fcaccportgrp-myFcHostPortGroup1" />
            </infraHPortS>
        </infraAccPortP>
        <infraNodeP name="myFcNode1">
            <infraLeafS name="myLeafSelector" type="range">
                <infraNodeBlk name="myLeaf104" from_="104" to_="104" />
            </infraLeafS>
            <infraRsAccPortP tDn="uni/infra/accportprof-myHostPorts" />
        </infraNodeP>
    </infraInfra>
```

```
</polUni>
```

Note
When this configuration is applied, a switch reload is required to bring up the ports as FC ports.

Currently only one contiguous range of ports can be converted to FC ports, and this range must be multiple of 4 ending with a port number that is multiple of 4. Examples are 1-4, 1-8, or 21-24.

Step 5 To create a FC uplink port interface policy and a policy group for uplink port channels, send a post with XML. This example executes the following requests:

- Creates a FC interface policy myFcUplinkIfPolicy2 for uplink ports. These are NP ports with trunking enabled.
- Creates a FC interface bundle policy group myFcUplinkBundleGroup2 that includes the FC uplink interface policy myFcUplinkIfPolicy2.
- Associates the policy group to the FC interface policy to convert these ports to FC ports.
- Creates an uplink port profile myFcUplinkPortProfile.
- Creates a port selector myFcUplinkSelector that specifies ports in range 1/9-12.
- Associates the host ports to the leaf node 104.

Example:

```
https://apic-ip-address/api/mo/uni.xml

<polUni>
    <infraInfra>
        <fcIfPol name="myFcUplinkIfPolicy2" portMode="np" trunkMode="trunk-on" speed="auto"/>
        <infraFuncP>
            <infraFcAccBndlGrp name="myFcUplinkBundleGroup2">
                <infraRsFcL2IfPol tnFcIfPolName="myFcUplinkIfPolicy2" />
            </infraFcAccBndlGrp>
        </infraFuncP>
        <infraAccPortP name="myFcUplinkPortProfile">
            <infraHPortS name="myFcUplinkSelector" type="range">
                <infraPortBlk name="myUplinkPorts" fromCard="1" toCard="1" fromPort="9" toPort="12"
 />
                <infraRsAccBaseGrp tDn="uni/infra/funcprof/fcaccportgrp-myFcUplinkBundleGroup2" />
            </infraHPortS>
        </infraAccPortP>
        <infraNodeP name="myFcNode1">
            <infraLeafS name="myLeafSelector" type="range">
                <infraNodeBlk name="myLeaf104" from_="104" to_="104" />
            </infraLeafS>
            <infraRsAccPortP tDn="uni/infra/accportprof-myUplinkPorts" />
        </infraNodeP>
    </infraInfra>
</polUni>
```

Note
When this configuration is applied, a switch reload is required to bring up the ports as FC ports.

Currently only one contiguous range of ports can be converted to FC ports, and this range must be multiple of 4 ending with a port number that is multiple of 4. Examples are 1-4, 1-8, or 21-24.

Step 6 To create the tenant, application profile, EPG and associate the FC bridge domain with the EPG, send a post with XML such as the following example. The example creates a bridge domain myFcBD1 under a target tenant configured to

support FC and an application EPG epg1. It associates the EPG with Fibre Channel domain myFcDomain1 and a Fibre Channel path to interface 1/7 on leaf switch 104. Each interface is associated with a VSAN.

Example:

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

<fvTenant name="tenant1">
  <fvCtx name="myFcVRF"/>
  <fvBD name="myFcBD1" type="fc">
    <fvRsCtx tnFvCtxName="myFcVRF"/>
  </fvBD>
  <fvAp name="app1">
    <fvAEPg name="epg1">
      <fvRsBd tnFvBDName="myFcBD1"/>
      <fvRsDomAtt tDn="uni/fc-myFcDomain1"/>
     <fvRsFcPathAtt tDn="topology/pod-1/paths-104/pathep-[fc1/1]" vsan="vsan-50" vsanMode="native"/>

     <fvRsFcPathAtt tDn="topology/pod-1/paths-104/pathep-[fc1/2]" vsan="vsan-50" vsanMode="native"/>

    </fvAEPg>
  </fvAp>
</fvTenant>
```

Step 7 To create a traffic map to pin server ports to uplink ports, send a post with XML such as the following example. The example creates a traffic map to pin server port vFC 1/47 to uplink port FC 1/7:

Example:

```
https://apic-ip-address/api/mo/uni/tn-tenant1.xml

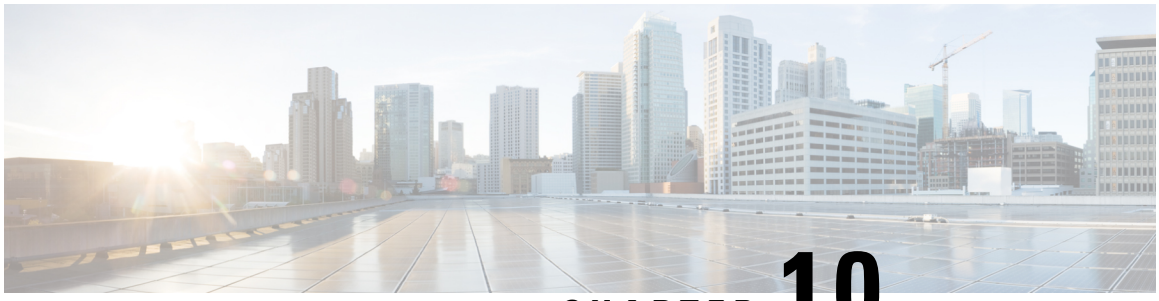
<fvTenant name="tenant1">
  <fvAp name="app1">
    <fvAEPg name="epg1">
     <fvRsFcPathAtt tDn="topology/pod-1/paths-104/pathep-[eth1/47]" vsan="vsan-50" vsanMode="native">

        <fcPinningLbl name="label1"/>
      </fvRsFcPathAtt>
    </fvAEPg>
  </fvAp>
</fvTenant>

https://apic-ip-address/api/mo/uni/tn-vfc_t1.xml

<fvTenant name="tenant1">
  <fcPinningP name="label1">
    <fcRsPinToPath tDn="topology/pod-1/paths-104/pathep-[fc1/7]"/>
  </fcPinningP>
</fvTenant>
```

Note
If traffic map pinning is configured for the first time, the server host port must be shut before configuring the first traffic map.



CHAPTER 10

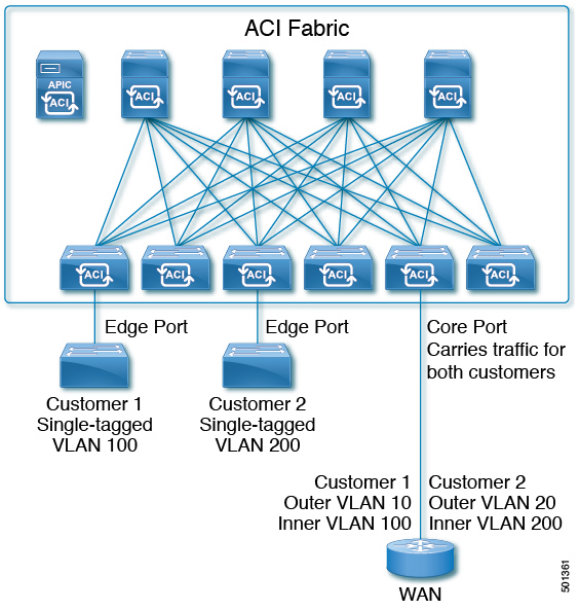
802.1Q Tunnels

This chapter contains the following sections:

- About ACI 802.1Q Tunnels, on page 179
- Configuring 802.1Q Tunnels Using the GUI, on page 181
- Configuring 802.1Q Tunnels Using the NX-OS Style CLI, on page 183

About ACI 802.1Q Tunnels

Figure 33: ACI 802.1Q Tunnels



You can configure 802.1Q tunnels on edge (tunnel) ports to enable point-to-multi-point tunneling of Ethernet frames in the fabric, with Quality of Service (QoS) priority settings. A Dot1q tunnel transports untagged, 802.1Q tagged, and 802.1ad double-tagged frames as-is across the fabric. Each tunnel carries the traffic from a single customer and is associated with a single bridge domain. Cisco Application Centric Infrastructure (ACI) front panel ports can be part of a Dot1q tunnel. Layer 2 switching is done based on the destination MAC (DMAC) and regular MAC learning is done in the tunnel. Edge port Dot1q tunnels are supported on Cisco Nexus 9000 series switches with "EX" or later suffixes in the switch model name.

You can configure multiple 802.1Q tunnels on the same core port to carry double-tagged traffic from multiple customers, each distinguished with an access encapsulation configured for each 802.1Q tunnel. You can also disable MAC address learning on 802.1Q tunnels. Both edge ports and core ports can belong to an 802.1Q tunnel with access encapsulation and disabled MAC address learning. Both edge ports and core ports in Dot1q tunnel are supported on Cisco Nexus 9000 series switches with "FX" or later suffixes in the switch model name.

IGMP and MLD packets can be forwarded through 802.1Q tunnels.

Terms used in this document may be different in the **Cisco Nexus 9000 Series** documents.

Table 7: 802.1Q Tunnel Terminology

ACI Documents	Cisco Nexus 9000 Series Documents
Edge Port	Tunnel Port
Core Port	Trunk Port

The following guidelines and restrictions apply:

- Layer 2 tunneling of VTP, CDP, LACP, LLDP, and STP protocols is supported with the following restrictions:
  - Link Aggregation Control Protocol (LACP) tunneling functions as expected only with point-to-point tunnels using individual leaf interfaces. It is not supported on port channels (PCs) or virtual port channels (vPCs).
  - CDP and LLDP tunneling with PCs or vPCs is not deterministic; it depends on the link it chooses as the traffic destination.
  - To use VTP for Layer 2 protocol tunneling, CDP must be enabled on the tunnel.
  - STP is not supported in an 802.1Q tunnel bridge domain when Layer 2 protocol tunneling is enabled and the bridge domain is deployed on Dot1q tunnel core ports.
  - Cisco ACI leaf switches react to STP TCN packets by flushing the end points in the tunnel bridge domain and flooding them in the bridge domain.
  - CDP and LLDP tunneling with more than two interfaces flood packets on all interfaces.
  - The destination MAC address of Layer 2 protocol packets tunneled from edge to core ports is rewritten as 01-00-0c-cd-cd-d0 and the destination MAC address of Layer 2 protocol packets tunneled from core to edge ports is rewritten with the standard default MAC address for the protocol.
- If a PC or vPC is the only interface in a Dot1q tunnel and it is deleted and reconfigured, remove the association of the PC/VPC to the Dot1q tunnel and reconfigure it.
- For 802.1Q tunnels deployed on switches that have EX in the product ID, Ethertype combinations of 0x8100+0x8100, 0x8100+0x88a8, 0x88a8+0x8100, and 0x88a8+0x88a8 for the first two VLAN tags are not supported.

  If the tunnels are deployed on a combination of EX and FX or later switches, then this restriction still applies.

  If the tunnels are deployed only on switches that have FX or later in the product ID, then this restriction does not apply.

- For core ports, the Ethertypes for double-tagged frames must be 0x8100 followed by 0x8100.
- You can include multiple edge ports and core ports (even across leaf switches) in a Dot1q tunnel.
- An edge port may only be part of one tunnel, but a core port can belong to multiple Dot1q tunnels.
- Regular EPGs can be deployed on core ports that are used in 802.1Q tunnels.
- L3Outs are not supported on interfaces enabled for Dot1q tunnel.
- FEX interfaces are not supported as members of a Dot1q tunnel.
- Interfaces configured as breakout ports do not support 802.1Q tunnels.
- Interface-level statistics are supported for interfaces in Dot1q tunnel, but statistics at the tunnel level are not supported.

Configuring 802.1Q Tunnels Using the GUI

Configuring 802.1Q Tunnel Interfaces Using the APIC GUI

Configure the interfaces that will use the tunnel, with the following steps:

Before you begin

Create the tenant that will be using the tunnel.

Procedure

Step 1 On the menu bar, click **Fabric** > **Access Policies**.

Step 2 On the Navigation bar, click **Policies** > **Interface** > **L2 Interface**.

Step 3 Right-click **L2 Interface**, select **Create L2 Interface Policy**, and perform the following actions:

a) In the **Name** field, type a name for the Layer 2 Interface policy.
b) Optional. Add a description of the policy. We recommended that you describe the purpose for the L2 Interface Policy.
c) To create an interface policy that enables an interface to be used as an edge port in a **Dot1q Tunnel**, in the **QinQ** field, click **edgePort**.
d) To create an interface policy that enables an interface to be used as a core port in **Dot1q Tunnels**, in the **QinQ** field, click **corePort**.

Step 4 Apply the L2 Interface policy to a Policy Group with the following steps:

a) Click on **Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** and expand **Policy Groups**.
b) Right-click **Leaf Access Port**, **PC Interface**, or **VPC Interface** and choose one of the following, depending on the type of interface you are configuring for the tunnel.

- **Create Leaf Access Port Policy Group**
- **Create PC Policy Group**
- **Create VPC Policy Group**

c) In the resulting dialog box, perform the following actions:

- In the **Name** field, type a name for the policy group.

  Optional. Add a description of the policy group. We recommend that you describe the purpose of the policy group.

- In the **L2 Interface Policy** field, click on the down-arrow and choose the L2 Interface Policy that you previously created.
- If you are tunneling the CDP Layer 2 Tunneled Protocol, click on the **CDP Policy** down-arrow, and in the policy dialog box add a name for the policy, disable the Admin State and click **Submit.**.
- If you are tunneling the LLDP Layer 2 Tunneled Protocol, click on the **LLDP Policy** down-arrow, and in the policy dialog box add a name for the policy, disable the Transmit State and click **Submit.**
- Click **Submit**.

Step 5 Create a Leaf Interface Profile with the following steps:

a) Click on **Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Profiles**.
b) Right-click on **Profiles**, select **Create Leaf Interface Profile**, and perform the following steps:

- In the **Name** field, type a name for the **Leaf Interface Profile**.

  Optional. Add a description.

- In the **Interface Selectors** field, click the **+**, and enter the following information:
  - In the **Name** field, type a name for the interface selector.

    Optional. Add a description.

  - In the **Interface IDs** field, enter the **Dot1q Tunnel** interface or multiple interfaces to be included in the tunnel.
  - In the **Interface Policy Group** field, click on the down arrow and select the interface policy group that you previously created .

Step 6 To create a static binding of the tunnel configuration to a port, click on **Tenant** > **Networking** > **Dot1Q Tunnels**. Expand **Dot1Q Tunnels** and click on the **Dot1Q Tunnels** ***policy_name*** perviously created and perform the following actions:

a) Expand the **Static Bindings** table to open **Create Static Binding** dialog box.
b) In the **Port** field, select the type of port.
c) In the **Node** field, select a node from the drop-down.
d) In the **Path** field, select the interface path from the drop-down and click **Submit**.

Configuring 802.1Q Tunnels Using the NX-OS Style CLI

Configuring 802.1Q Tunnels Using the NX-OS Style CLI

Note You can use ports, port-channels, or virtual port channels for interfaces included in a **Dot1q Tunnel**. Detailed steps are included for configuring ports. See the examples below for the commands to configure edge and core port-channels and virtual port channels.

Create a **Dot1q Tunnel** and configure the interfaces for use in the tunnel using the NX-OS Style CLI, with the following steps:

Note **Dot1q Tunnels** must include 2 or more interfaces. Repeat the steps (or configure two interfaces together), to mark each interface for use in a **Dot1q Tunnel**. In this example, two interfaces are configured as edge-switch ports, used by a single customer.

Use the following steps to configure a **Dot1q Tunnel** using the NX-OS style CLI:

1. Configure at least two interfaces for use in the tunnel.
2. Create a **Dot1q Tunnel**.
3. Associate all the interfaces with the tunnel.

Before you begin

Configure the tenant that will use the **Dot1q Tunnel**.

SUMMARY STEPS

1. **configure**
2. Configure two interfaces for use in an 802.1Q tunnel, with the following steps:
3. **leaf** *ID*
4. **interface ethernet** *slot/port*
5. **switchport mode dot1q-tunnel** {**edgePort** | **corePort**}
6. Create an 802.1Q tunnel with the following steps:
7. **leaf** *ID*
8. **interface ethernet***slot/port*
9. **switchport tenant***tenant-name***dot1q-tunnel** *tunnel-name*
10. Repeat steps 7 to 10 to associate other interfaces with the tunnel.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters configuration mode.
Step 2	Configure two interfaces for use in an 802.1Q tunnel, with the following steps:	
Step 3	**leaf** *ID* **Example:** `apic1(config)# leaf 101`	Identifies the leaf where the interfaces of the **Dot1q Tunnel** will be located.
Step 4	**interface ethernet** *slot/port* **Example:** `apic1(config-leaf)# interface ethernet 1/13-14`	Identifies the interface or interfaces to be marked as ports in a tunnel.

Step 5	**switchport mode dot1q-tunnel** {**edgePort** | **corePort**}
	Example:
	`apic1(config-leaf-if)# switchport mode dot1q-tunnel edgePort`
	`apic1(config-leaf-if)# exit`
	`apic1(config-leaf)# exit`
	`apic1(config)# exit`
		Marks the interfaces for use in an 802.1Q tunnel, and then leaves the configuration mode.
		The example shows configuring some interfaces for edge port use. Repeat steps 3 to 5 to configure more interfaces for the tunnel.

Step 6	Create an 802.1Q tunnel with the following steps:	
Step 7	**leaf** *ID* **Example:** `apic1(config)# leaf 101`	Returns to the leaf where the interfaces are located.
Step 8	**interface ethernet***slot/port* **Example:** `apic1(config-leaf)# interface ethernet 1/13-14`	Returns to the interfaces included in the tunnel.

Step 9	**switchport tenant***tenant-name***dot1q-tunnel** *tunnel-name*
	Example:
	`apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_edgetunnel`
	`apic1(config-leaf-if)# exit`
		Associates the interfaces to the tunnel and exits the configuration mode.

Step 10	Repeat steps 7 to 10 to associate other interfaces with the tunnel.	

Example: Configuring an 802.1Q Tunnel Using Ports with the NX-OS Style CLI

The example marks two ports as edge port interfaces to be used in a **Dot1q Tunnel**, marks two more ports to be used as core port interfaces, creates the tunnel, and associates the ports with the tunnel.

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/13-14
apic1(config-leaf-if)# switchport mode dot1q-tunnel edgePort
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)leaf 102
apic1(config-leaf)# interface ethernet 1/10, 1/21
apic1(config-leaf-if)# switchport mode dot1q-tunnel corePort
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# tenant tenant64
apic1(config-tenant)# dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# l2protocol-tunnel cdp
apic1(config-tenant-tunnel)# l2protocol-tunnel lldp
apic1(config-tenant-tunnel)#  access-encap 200
apic1(config-tenant-tunnel)# mac-learning disable
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/13-14
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 1/10, 1/21
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
```

Example: Configuring an 802.1Q Tunnel Using Port-Channels with the NX-OS Style CLI

The example marks two port-channels as edge-port 802.1Q interfaces, marks two more port-channels as core-port 802.1Q interfaces, creates a **Dot1q Tunnel**, and associates the port-channels with the tunnel.

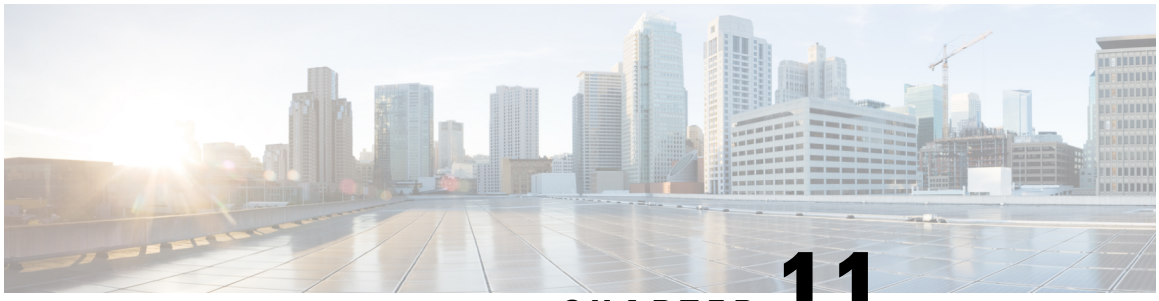
```
apic1# configure
apic1(config)# tenant tenant64
apic1(config-tenant)# dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# l2protocol-tunnel cdp
apic1(config-tenant-tunnel)# l2protocol-tunnel lldp
apic1(config-tenant-tunnel)#  access-encap 200
apic1(config-tenant-tunnel)# mac-learning disable
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 101
apic1(config-leaf)# interface port-channel pc1
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/2-3
apic1(config-leaf-if)# channel-group pc1
```

```
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface port-channel pc1
apic1(config-leaf-if)# switchport mode dot1q-tunnel edgePort
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 102
apic1(config-leaf)# interface port-channel pc2
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/4-5
apic1(config-leaf-if)# channel-group pc2
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface port-channel pc2
apic1(config-leaf-if)# switchport mode dot1q-tunnel corePort
apic1(config-leaf-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
```

Example: Configuring an 802.1Q Tunnel Using Virtual Port-Channels with the NX-OS Style CLI

The example marks two virtual port-channels (vPCs) as edge-port 802.1Q interfaces for the**Dot1q Tunnel**, marks two more vPCs as core-port interfaces for the tunnel, creates the tunnel, and associates the virtual port-channels with the tunnel.

```
apic1# configure
apic1(config)# vpc domain explicit 1 leaf 101 102
apic1(config)# vpc context leaf 101 102
apic1(config-vpc)# interface vpc vpc1
apic1(config-vpc-if)# switchport mode dot1q-tunnel edgePort
apic1(config-vpc-if)# exit
apic1(config-vpc)# exit
apic1(config)# vpc domain explicit 1 leaf 103 104
apic1(config)# vpc context leaf 103 104
apic1(config-vpc)# interface vpc vpc2
apic1(config-vpc-if)# switchport mode dot1q-tunnel corePort
apic1(config-vpc-if)# exit
apic1(config-vpc)# exit
apic1(config)# tenant tenant64
apic1(config-tenant)# dot1q-tunnel vrf64_tunnel
apic1(config-tenant-tunnel)# l2protocol-tunnel cdp
apic1(config-tenant-tunnel)# l2protocol-tunnel lldp
apic1(config-tenant-tunnel)#  access-encap 200
apic1(config-tenant-tunnel)# mac-learning disable
apic1(config-tenant-tunnel)# exit
apic1(config-tenant)# exit
apic1(config)# leaf 103
apic1(config-leaf)# interface ethernet 1/6
apic1(config-leaf-if)# channel-group vpc1 vpc
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config)# leaf 104
apic1(config-leaf)# interface ethernet 1/6
apic1(config-leaf-if)# channel-group vpc1 vpc
apic1(config-leaf-if)# exit
apic1(config-leaf)# exit
apic1(config-vpc)# interface vpc vpc1
apic1(config-vpc-if)# switchport tenant tenant64 dot1q-tunnel vrf64_tunnel
apic1(config-vpc-if)# exit
```



CHAPTER 11

Q-in-Q Encapsulation Mapping for EPGs

- Q-in-Q Encapsulation Mapping for EPGs, on page 187
- Configuring Q-in-Q Encapsulation Mapping for EPGs Using the GUI, on page 188
- Mapping EPGs to Q-in-Q Encapsulated Leaf Interfaces Using the NX-OS Style CLI, on page 191

Q-in-Q Encapsulation Mapping for EPGs

Using Cisco Application Policy Infrastructure Controller (APIC), you can map double-tagged VLAN traffic ingressing on a regular interface, PC, or vPC to an EPG. When this feature is enabled, when double-tagged traffic enters the network for an EPG, both tags are processed individually in the fabric and restored to double-tags when egressing the Cisco Application Centric Infrastructure (ACI) switch. Ingressing single-tagged and untagged traffic is dropped.

The following guidelines and limitations apply:

- This feature is only supported on Cisco Nexus 9300-FX platform switches.
- Both the outer and inner tag must be of EtherType 0x8100.
- MAC learning and routing are based on the EPG port, sclass, and VRF instance, not on the access encapsulations.
- QoS priority settings are supported, derived from the outer tag on ingress, and rewritten to both tags on egress.
- EPGs can simultaneously be associated with other interfaces on a leaf switch, that are configured for single-tagged VLANs.
- Service graphs are supported for provider and consumer EPGs that are mapped to Q-in-Q encapsulated interfaces. You can insert service graphs, as long as the ingress and egress traffic on the service nodes is in single-tagged encapsulated frames.
- When vPC ports are enabled for Q-in-Q encapsulation mode, VLAN consistency checks are not performed.

The following features and options are not supported with this feature:

- Per-port VLAN feature
- FEX connections
- Mixed mode

For example, an interface in Q-in-Q encapsulation mode can have a static path binding to an EPG with double-tagged encapsulation only, not with regular VLAN encapsulation.

- STP and the "Flood in Encapsulation" option
- Untagged and 802.1p mode
- Multi-pod and Multi-Site
- Legacy bridge domain
- L2Out and L3Out connections
- VMM integration
- Changing a port mode from routed to Q-in-Q encapsulation mode
- Per-VLAN mis-cabling protocol on ports in Q-in-Q encapsulation mode

Configuring Q-in-Q Encapsulation Mapping for EPGs Using the GUI

Enabling Q-in-Q Encapsulation on Specific Leaf Switch Interfaces Using the GUI

Leaf switch ports, PCs, or vPCs are enabled for Q-in-Q encapsulation mode in the **Interface** tab of one of the following locations in the APIC GUI.

- **Fabric** > **Inventory** > **Topology**
- **Fabric** > **Inventory** > **Pod**
- **Fabric** > **Inventory** > **Pod** > ***leaf-name***

Configure vPCs on the **Topology** or **Pod Interface** tab.

Before you begin

The tenant, application profile, and the application EPG that will be mapped with an interface configured for Q-in-Q mode should be created.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Inventory** and click **Topology**, **Pod**, or expand **Pod** and choose a leaf.

Step 2 On the **Topology** or **Pod** panel **Interface** tab.

Step 3 Click the **Operation**/**Configuration** toggle-button to display the configuration panel.

Step 4 Click + to add diagrams of leaf switches, choose one or more switches, and click **Add Selected**.

On the ***leaf-name*** panel **Interface** tab, a diagram of the switch appears automatically, after you click the **Operation**/**Configuration**toggle-button.

Step 5 Click the interfaces to be enabled for Q-in-Q encapsulation mode.

Step 6 To configure a port, perform the following steps:

a) Click **L2** on the upper left.
b) On the L2 tab, on the **L2 QinQ State** field, click **Double Q Tag Port** and click **Submit**

Step 7 To configure a PC, perform the following steps:

a) Click **PC** on the upper left.
b) On the **Physical Interface** tab, enter the **Policy Group Name**.
c) On the L2 tab, on the **L2 QinQ State** field, click **Double Q Tag Port** and click **Submit**

Step 8 To configure a vPC, perform the following steps:

a) On two leaf switch diagrams, click the interfaces for the two legs of the VPC.
b) Click **vPC**.
c) On the **Physical Interface** tab, enter the **Logical Pair ID** (The identifier for the auto-protection group. Each protection group has a unique ID. The ID is a range of 1 to 1000) and the **Policy Group Name**.
d) On the L2 tab, on the **L2 QinQ State** field, click **Double Q Tag Port** and click **Submit**

Enabling Q-in-Q Encapsulation for Leaf Interfaces With Fabric Interface Policies Using the GUI

Enable leaf interfaces, PCs, and vPCs for Q-in-Q encapsulation, using a leaf interface profile.

Before you begin

The tenant, application profile, and the application EPG that will be mapped with an interface configured for Q-in-Q mode should be created.

Procedure

Step 1 On the menu bar, click **Fabric** > **External Access Policies**.

Step 2 On the Navigation bar, click **Policies** > **Interface** > **L2 Interface**.

Step 3 Right-click **L2 Interface**, select **Create L2 Interface Policy**, and perform the following actions:

a) In the **Name** field, enter a name for the Layer 2 Interface policy.
b) Optional. Add a description of the policy. We recommend that you describe the purpose for the L2 Interface Policy.
c) To create an interface policy that enables Q-in-Q encapsulation, in the **QinQ** field, click **doubleQtagPort**.
d) Click **Submit**.

Step 4 Apply the L2 Interface policy to a Policy Group with the following steps:

a) Click on **Fabric** > **External Access Policies** > **Interfaces** > **Leaf Interfaces**, and expand **Policy Groups**.
b) Right-click **Leaf Access Port**, **PC Interface**, or **vPC Interface** and choose one of the following, depending on the type of interface you are configuring for the tunnel.

- **Create Leaf Access Port Policy Group**

- **Create PC Policy Group**
- **Create vPC Policy Group**

c) In the resulting dialog box, enter the policy group name, choose the L2 Interface policy that you previously created, and click **Submit.**

Step 5 Create a Leaf Interface Profile with the following steps:

a) Click on **Fabric** > **External Access Policies** > **Interface** > **Leaf Interfaces** > **Profiles**.

b) Right-click on **Leaf Profiles**, choose **Create Leaf Interface Policy**, and perform the following steps:

- In the **Name** field, type a name for the **Leaf Interface Profile**.

  Optional. Add a description.

- On the **Interface Selectors** field, click the +, and enter the following information:
  - In the **Name** field, type a name for the interface selector.

    Optional. Add a description.
  - Enter the selector name, and optionally, a description.
  - In the Interface IDs field, enter the interface or multiple interfaces to be included in the profile.
  - In the **Interface Policy Group** field, choose the interface policy group that you previously created.

Mapping an EPG to a Q-in-Q Encapsulation-Enabled Interface Using the GUI

You can associate EPGs with Q-in-Q encapsulation-enabled interfaces in one of the following models:

- Deploy a static EPG on specific Q-in-Q encapsulation-enabled interfaces
- Statically link an EPG with a Q-in-Q encapsulation-enabled leaf switch
- Associate an EPG with a Q-in-Q encapsulation-enabled endpoint (with a static MAC address)

All three tasks are performed in the same area of the APIC GUI.

Before you begin

- Create the tenant, application profile, and application EPG that will be mapped with an interface configured for Q-in-Q mode.
- The target interfaces should be configured for Q-in-Q encapsulation.

SUMMARY STEPS

1. In the menu bar, click **Tenants** > ***tenant-name***.
2. In the Navigation pane, expand **Application Profiles** > > ***application-profile-name*** > **Application EPGs** > ***application-EPG-name***.
3. To deploy a static EPG on an interface, PC, or vPC that has been enabled for Q-in-Q mode, perform the following steps:

4. To statically link an EPG with a node enabled with Q-in-Q mode, perform the following steps:
5. To associate an EPG with a static endpoint, perform the following steps:

DETAILED STEPS

Procedure

Step 1 In the menu bar, click **Tenants** > ***tenant-name***.

Step 2 In the Navigation pane, expand **Application Profiles** > > ***application-profile-name*** > **Application EPGs** > ***application-EPG-name***.

Step 3 To deploy a static EPG on an interface, PC, or vPC that has been enabled for Q-in-Q mode, perform the following steps:

a) Under the application EPG, right- click **Static Ports** and choose **Deploy Static EPG on PC, vPC, or Interface**.
b) Choose the path type, the node, and the path to the Q-in-Q enabled interface.
c) On the **Port Encap (or Secondary VLAN for Micro-Seg)** field, choose **QinQ** and enter the outer and inner VLAN tags for traffic mapped to the EPG.
d) Click **Submit**.

Step 4 To statically link an EPG with a node enabled with Q-in-Q mode, perform the following steps:

a) Under the application EPG, right- click **Static Leafs** and choose **Statically Link With Node**.
b) In the Node field, choose the Q-in-Q-enabled switches from the list.
c) On the Encap field, choose **QinQ** and enter the outer and inner VLAN tags for the EPG.
d) Click **Submit**.

Step 5 To associate an EPG with a static endpoint, perform the following steps:

a) Under the application EPG, right- click **Static EndPoints** and choose **Create Static EndPoint**.
b) Enter the MAC address of the interface.
c) Choose the path type, node, and path to the Q-in-Q encapsulation-enabled interface.
d) Optional. Add IP addresses for the endpoint.
e) On the **Encap** field, choose **QinQ** and enter the outer and inner VLAN tags.
f) Click **Submit**.

Mapping EPGs to Q-in-Q Encapsulated Leaf Interfaces Using the NX-OS Style CLI

Enable an interface for Q-in-Q encapsulation and associate the interface with an EPG.

Before you begin

Create the tenant, application profile, and application EPG that will be mapped with an interface configured for Q-in-Q mode.

SUMMARY STEPS

1. **Configure**

2. **leaf** *number*
3. **interface ethernet***slot/port*
4. **switchport mode dot1q-tunnel** doubleQtagPort
5. **switchport trunkqinq outer-vlan***vlan-number* **inner-vlan** *vlan-number* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name*

DETAILED STEPS

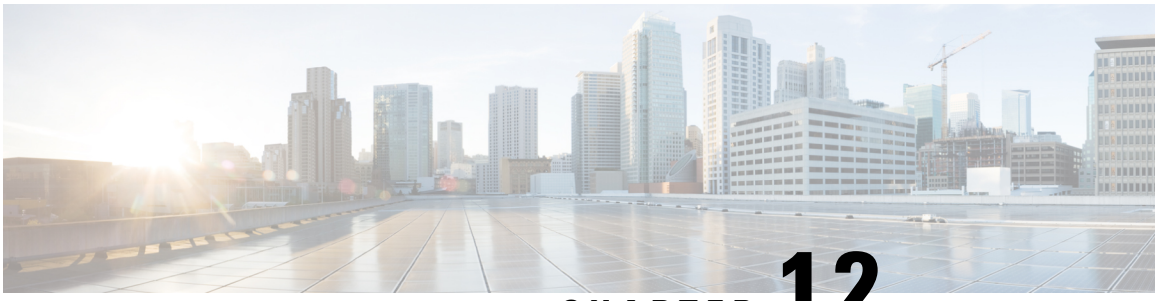
Procedure

	Command or Action	Purpose
Step 1	**Configure** **Example:** `apic1# configure`	Enters global configuration mode.
Step 2	**leaf** *number* **Example:** apic1(config)# leaf 101	Specifies the leaf to be configured.
Step 3	**interface ethernet***slot/port* **Example:** `apic1 (config-leaf)# interface ethernet 1/25`	Specifies the interface to be configured.
Step 4	**switchport mode dot1q-tunnel** doubleQtagPort **Example:** `apic1(config-leaf-if)# switchport mode dot1q-tunnel doubleQtagPort`	Enables an interface for Q-in-Q encapsulation.
Step 5	**switchport trunkqinq outer-vlan***vlan-number* **inner-vlan** *vlan-number* **tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **Example:** `apic1(config-leaf-if)# switchport trunk qinq outer-vlan 202 inner-vlan 203 tenant tenant64 application AP64 epg EPG64`	Associates the interface with an EPG.

Example

The following example enables Q-in-Q encapsulation (with outer-VLAN ID 202 and inner-VLAN ID 203) on the leaf interface 101/1/25, and associates the interface with EPG64.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/25
apic1(config-leaf-if)#switchport mode dot1q-tunnel doubleQtagPort
apic1(config-leaf-if)# switchport trunk qinq outer-vlan 202 inner-vlan 203 tenant tenant64
 application AP64 epg EPG64
```



CHAPTER 12

Breakout Ports

This chapter contains the following sections:

- Configuration of Breakout Ports, on page 193
- Guidelines and Limitations for Dynamic Breakout Ports on Down Links, on page 194
- Guidelines and Limitations for Auto Breakout Ports on Fabric Links, on page 198
- Configuring Breakout Ports With Profiles and Selectors Using the GUI, on page 200
- Configuring Breakout Ports With Profiles and Selectors Using the GUI, on page 203
- Configuring Breakout Ports With the Interface Configuration Using the GUI, on page 206
- Configuring Dynamic Breakout Ports Using the NX-OS Style CLI, on page 206

Configuration of Breakout Ports

Breakout cables are suitable for very short links and offer a cost effective way to connect within racks and across adjacent racks. Breakout enables a 40 Gigabit (Gb) port to be split into four independent and logical 10Gb ports, a 100Gb port to be split into four independent and logical 25Gb ports, or a 400Gb port to be split into four independent and logical 100Gb ports.

You configure breakout on the down links (also known as the access-facing ports or downlink ports) and fabric links of the switches. Fabric links form the connections between the leaf switches and spine switches, or between the tier 1 leaf switches and tier 2 leaf switches for a multi-tier topology.

You can configure breakout ports in the following ways:

- You can use port profiles and selectors. With this method, you configure a breakout leaf port with an leaf interface profile, associate the profile with a switch, and configure the sub-ports.
- Beginning with the Cisco Application Policy Infrastructure Controller (APIC) 6.0(1) release, you can use the **Fabric** > **Access Policies** > **Interface Configuration** workflow.
- You can use the **Fabric** > **Inventory** > ***pod*** > ***leaf_name*** workflow. Beginning with the Cisco APIC 6.0(1) release, the inventory view configuration also uses the interface configuration.

Guidelines and Limitations for Dynamic Breakout Ports on Down Links

The 40Gb to 10Gb dynamic breakout feature is supported on the access-facing ports of the following switches:

- N9K-C93180LC-EX
- N9K-C93180YC-FX
- N9K-C9336C-FX2
- N9K-C93360YC-FX2
- N9K-C93216TC-FX2
- N9K-C93108TC-FX3P
- N9K-C93180YC-FX3
- N9K-C93600CD-GX
- N9K-C9364C-GX
- N9K-C9408 (beginning in the 6.0(2) release)
- N9K-C9348D-GX2A (beginning in the 6.0(3) release)
- N9K-C9364D-GX2A (beginning in the 6.0(3) release)
- N9K-C9332D-GX2B (beginning in the 6.0(3) release)

The 100Gb to 25Gb breakout feature is supported on the access ports of the following switches:

- N9K-C93180LC-EX
- N9K-C9336C-FX2
- N9K-C93180YC-FX
- N9K-C93360YC-FX2
- N9K-C93216TC-FX2
- N9K-C93108TC-FX3P
- N9K-C93180YC-FX3
- N9K-C93600CD-GX
- N9K-C9364C-GX
- N9K-C9408 (beginning in the 6.0(2) release)
- N9K-C9348D-GX2A (beginning in the 6.0(3) release)
- N9K-C9364D-GX2A (beginning in the 6.0(3) release)
- N9K-C9332D-GX2B (beginning in the 6.0(3) release)

The 400Gb to 100Gb breakout feature is supported on the access ports of the following switches:

- N9K-C9348D-GX2A
- N9K-C9364D-GX2A
- N9K-C9332D-GX2B
- N9K-C93600CD-GX
- N9K-C9316D-GX
- N9K-C9408 (beginning in the 6.0(2) release)
- Beginning with the 6.0(2) release, the QDD-400G-SR4.2-BD optic is supported on 400Gb ports. The peer node with the 100Gb speed must use QSFP-100G-SR1.2 optics.
- Beginning with the 6.0(3) release, 400Gb-to-100Gb breakout using a QDD-4ZQ100 cable of any length is supported only with the following platforms:
  - N9K-C9332D-GX2B, N9K-C9348D-GX2A, N9K-C9364D-GX2A, N9K-C9316D-GX, and N9K-C93600CD-GX leaf and spine switches
  - N9K-X9716D-GX line card
- The N9K-C9400-SW-GX2A switch supports 400Gb breakout with the N9K-X9400-8D line card expansion module and the peer 100G-PCC is supported with the N9K-X9400-8D and N9K-X9400-16W line card expansion modules.
- QDD-400G-DR4-S, QDD-4X100G-FR-S, QDD-4X100G-LR-S optics are supported on 400Gb ports. The peer node with the 100Gb speed can use the following optics:
  - QSFP-100G-DR-S
  - QSFP-100G-FR-S
  - QSFP-100G-LR-S

Before you configure breakout ports, connect a 40Gb port to four 10Gb ports, a 100Gb port to four 25Gb ports, or a 400Gb port to four 100Gb ports with one of the following cables:

- Cisco QSFP-4SFP10G

  Beginning in the 6.0(3) release, GX2 switches support this direct-attach cable.
- Cisco QSFP-4SFP25G

  Beginning in the 6.0(3) release, GX2 switches support this direct-attach cable.
- Cisco QSFP-4X10G-AOC

  Beginning in the 6.0(3) release, GX2 switches support this direct-attach cable.
- Cisco QDD-4ZQ100-CU (1M, 2M, 2.5M, and 3M) (beginning in the 6.0(3) release)
- MPO to breakout splitter cable with QSFP-40G-SR4 and 4 x SFP-10G-SR on the ends
- MPO to breakout splitter cable with QSFP-100G-SR4-S and 4 x SFP-25G-SR-S on the ends
- MPO to breakout splitter cable with QDD-400G-DR4-S, QDD-4X100G-FR-S, or QDD-4X100G-LR-S, and 4 x QSFP-100G-DR-S, 4 x QSFP-100G-FR-S, or 4 x QSFP-100G-LR-S on the ends

- MPO to breakout splitter cable with QDD-400G-SR4.2-BD and 4 x QSFP-100G-SR1.2 on the ends.



Note For the supported optics and cables, see the *Cisco Optics-to-Device Compatibility Matrix*:

https://tmgmatrix.cisco.com/

Observe the following guidelines and limitations:

- Breakout ports are supported on both down links and converted down links.
- The following switches support dynamic breakouts (both 100Gb and 40Gb) on profiled QSFP ports:
  - Cisco N9K-C93180YC-FX
  - Cisco N9K-C93216TC-FX2
  - Cisco N9K-C93360YC-FX2
  - Cisco N9K-C93600CD-GX

    This applies only to ports 1/25 to 34. Ports 1/29 to 34 can be used for dynamic breakouts if the ports are converted to down links.
  - Cisco N9K-C9336C-FX2

    You can configure up to 34 dynamic breakouts.
  - Cisco N9K-C9364C-GX

    You can configure up to 30 dynamic breakouts on odd-numbered profiled QSFP ports from 1/1 to 59.
  - Cisco N9K-93600CD-GX

    You can configure up to 12 dynamic breakouts out of 24 40/100G ports and up to 10 dynamic breakouts out of ports 25 to 34. Ports 29 to 34 can be breakouts if the ports are converted to down links. The last 2 ports (ports 35 and 36) are reserved for fabric links.
  - Cisco N9K-C9400-SW-GX2A with the Cisco N9K-X9400-16W linecard

    You can configure breakouts on odd-numbered profiled QSFP ports.
- The Cisco N9K-C9336C-FX2 switch supports LACP fast hello on the breakout sub-port.
- Breakout ports cannot be used for Cisco Application Policy Infrastructure Controller (APIC) connectivity.
- Dynamic breakout configuration on spine switch interfaces is not supported.
- Fast Link Failover policies are not supported on the same port with the dynamic breakout feature.
- Breakout subports can be used in the same way other port types in the policy model are used.
- When a port is enabled for dynamic breakout, other policies (except monitoring policies) on the parent port are no longer valid.
- When a port is enabled for dynamic breakout, other EPG deployments on the parent port are no longer valid.

- A breakout sub-port can not be further broken out using a breakout policy group.
- Dynamic breakout or breakout of a 400Gb port to four 100Gb ports configured using a Cisco APIC policy is supported on QDD-4X100G-FR-S and QDD-4X100G-LR-S optics.
- A breakout sub-port supports LACP. The LACP transmit rate configuration defined in the "default" port channel member policy is used by default. The LACP transmit rate can be changed by either changing the "default" port channel member policy or using an override policy group on each PC/vPC interface policy group.
- Deleting the port profile policy on an already broken out port will clean up the breakout configuration and bring the port back up in the native direction upon switch reload.
- If the LACP transmit rate on port channels that have breakout sub-ports need to be changed, then all the port channels that include breakout sub-ports need to use the same LACP transmit rate configuration. You can configure an override policy to set the transmit rate as follows:
  1. Configure/change the default port channel member policy to include Fast Transmit Rate (**Fabric** > **Access Policies** > **Policies** > **Interface** > **Port Channel Member**).
  2. Configure all the PC/vPC interface policy groups to include the above default port channel member policy under the override policy groups (**Fabric** > **Access Policies** > **Interfaces** > **Leaf Interfaces** > **Policy Groups** > **PC/vPC Interface**).
- The following guidelines and limitations apply to the Cisco N9K-C9364C-GX switch:
  - Odd numbered ports (on rows 1 and row 3) support breakout. Adjacent even numbered ports (on row 2 or row 4) will be disabled ("hw-disabled"). This is applicable to ports 1/1 to 60.
  - The last 2 ports (1/63 and 64) are reserved for fabric links.
  - Ports 1/61 and 62 can be converted to down links, but breakout is not supported. Breakout ports and 40/100G non-breakout ports cannot be mixed in a set of 4 ports starting from 1/1, such as 1/1 to 4 or 1/5 to 8.

    For example, if port 1/1 is breakout enabled, port 1/3 can be used with breakout enabled or native 10G. Port 1/3 will be error-disabled if it is 40/100G.
  - The maximum number of down links are 30 x 4 ports 10/25 (breakout) + 2 ports (1/61 and 62) = 122 ports. Ports 1/63 and 64 are reserved for fabric links and even numbered ports from 1/2 to 60 are error-disabled.
  - This switch supports 10G with QSA on all ports. Native 10G requires QSA.
- The following guidelines and limitations apply to the Cisco N9K-93600CD-GX switch:
  - Odd numbered ports (all ports on row 1) support breakout. Even numbered ports on row 2 will be disabled ("hw-disabled"). This is applicable only to ports 1 to 24.
  - Breakout and 40/100G non-breakout cannot be mixed in a set of 4 ports starting from 1/1 until 1/24, such as 1/1 to 4 or 1/5 to 8. For example:
    - For ports 1/1 to 24, you can have 4 ports per set.

      For example, if port 1/1 is breakout enabled, port 1/3 can be used with breakout enabled or native 10G. Port 1/3 will be error-disabled if it is 40/100G.
    - For ports 1/25 to 28, you can have 2 ports per set.

For example, even if port 1/25 is breakout enabled, port 1/27 can be used with 40/100G.

- The maximum number of down links are 12 x 4 ports 10/25G (breakout) + 10 x 4 ports 10/25G (breakout) = 88 ports. Ports 35 and 36 are reserved for fabric links and 12 ports are disabled.
- This switch supports 10G with QSA on all ports. Native 10G requires QSA.

Guidelines and Limitations for Auto Breakout Ports on Fabric Links

When you insert a transceiver into a line card on which breakout is supported, the ports become broken out automatically. You do not need to configure breakout manually.

The 400G to 100G breakout feature is supported on the fabric ports of the following line card:

- N9K-X9716D-GX with the QDD-4X100G-FR-S, QDD-4X100G-LR-S, or QDD-400G-SR4.2-BD transceiver

The 400G to 100G breakout feature is supported on the fabric ports of the following switches:

- N9K-C9348D-GX2A
- N9K-C9364D-GX2A
- N9K-C9332D-GX2B
- N9K-C93600CD-GX
- N9K-C9316D-GX
- N9K-C9408 (beginning in the 6.0(2) release)
- Beginning with the 6.0(2) release, the QDD-400G-SR4.2-BD optic is supported on 400G ports. The peer node with the 100G speed must use QSFP-100G-SR1.2 optics.
- Beginning with the 6.0(3) release, 400G-to-100G breakout using a QDD-4ZQ100 cable of any length is supported only with the following platforms:
  - N9K-C9332D-GX2B, N9K-C9348D-GX2A, N9K-C9364D-GX2A, N9K-C9316D-GX, and N9K-C93600CD-GX leaf and spine switches
  - N9K-X9716D-GX line card
- The N9K-C9400-SW-GX2A switch supports 400G breakout with the N9K-X9400-8D line card expansion module and the peer 100G-PCC is supported with the N9K-X9400-8D and N9K-X9400-16W line card expansion modules.
- QDD-4X100G-FR-S and QDD-4X100G-LR-S optics are supported on 400G ports. The peer node with the 100G speed can use the following optics:
  - QSFP-100G-DR-S
  - QSFP-100G-FR-S
  - QSFP-100G-LR-S

Connect the ports with one of the following cables:

- Cisco QDD-4ZQ100-CU (1M, 2M, 2.5M, and 3M) (beginning in the 6.0(3) release)
- MPO to 4xLC breakout splitter cable with QDD-4X100G-FR-S or QDD-4X100G-LR-S and 4 x QSFP-100G-DR-S, 4 x QSFP-100G-FR-S, or 4 x QSFP-100G-LR-S on the ends
- MPO to breakout splitter cable with QDD-400G-SR4.2-BD and 4 x QSFP-100G-SR1.2 on the ends.

Observe the following general guidelines and limitations:

- Deleting the port profile policy on an already broken out port will clean up the breakout configuration and bring the port back up in the native direction upon switch reload.

Observe the following guidelines and limitations for 400G to 4x100G breakout on fabric links:

- To change a fabric port with a QDD-400G-SR4.2-BD optic from non-breakout to breakout, you must remove the transceiver, connect the breakout cable to the transceiver, then reinsert the transceiver.
- GX2 switches support the following switch to switch connectivity:
  - Spine switch-to-leaf switch
  - Leaf switch-to-spine switch
  - Leaf switch-to-leaf switch (multi-tier)
- GX line cards support the following switch to switch connectivity:
  - Spine switch to leaf switch
- Spine switch-to-spine switch breakout is not supported.
- Spine switch-to-IPN or ISN breakout is only supported with Auto Breakout and only with the following 100G and 400G optics.
  - QDD-4X100G-FR-S or QDD-4X100G-LR-S optic on the spine switch and the compatible 100G optics such as QSFP-100G-FR-S and QSFP-100G-LR-S on the IPN or ISN side.
  - QDD-400G-SR4.2-BD optic on the spine switch and the QSFP-100G-SR1.2 optic on the IPN or ISN side.
  - The link does not come up when QDD-400G-SR4.2-BD optic on both the sides.
- In a Cisco ACI environment, the operational requirements for the QDD-400G-SR4.2-BD transceiver vary significantly depending on whether the interface is configured as a Fabric or Access port and the specific nature of the peer device.

  1. Fabric Port Behavior

  Fabric ports utilize automated discovery mechanisms that differ from standard access ports.

  - ACI-to-ACI Connections:

    In native 400G mode, the system automatically performs a breakout upon detecting the transceiver and, following the successful exchange and validation of LLDP TLVs across all four subports, updates the configuration to transition the interface to native 400G mode.

To operate in breakout mode, the local device must be equipped with a QDD-400G-SR4.2-BD transceiver, while the peer device must utilize 4x 100G-SR1.2 transceivers.

- ACI-to-Non-ACI Connections:

  In native 400G mode, the link will fail to establish because the ACI fabric port attempts an auto-breakout that the non-ACI peer does not support, resulting in a known configuration mismatch.

  To operate in breakout mode, compatibility is ensured by using the QDD-400G-SR4.2-BD transceiver on the ACI side and 4x 100G-SR1.2 transceivers on the peer device, mirroring the requirements for ACI-to-ACI connections.

2. Access Port Behavior

   Access ports do not rely on the same automated fabric discovery as fabric ports and require explicit configuration.

   - In native 400G mode, a standard QDD-400G-SR4.2-BD connection is required between both devices.

     In breakout mode, the ACI access port requires the manual application of a 4x 100G breakout policy to correctly initialize the interface.

- If you downgrade to a release that does not support breakout for the given hardware and ports, the breakout ports will become not broken out and the links will be down. If all connections between a spine and leaf switch are only breakout, then upon downgrading to a release that does not support breakout, the links will go down and the node will be out of the fabric.
- If you remove an SFP transceiver from a switch, you must wait at least 15 seconds before re-adding the transceiver.
- With the Cisco Nexus 9300 GX2 series or Cisco N9K-X9716D-GX line cards, ports will not come up if an optic is replaced while the line card is powered off. For example:
  1. You have a Cisco N9K-X9716D-GX line card in slot 4 with a 4x100-FR-S transceiver inserted in a port (for example, port 8). Port 8 is in broken out into four ports (Eth4/8/1-4) due to the auto-breakout feature that automatically activates when a 4x100-FR-S transceiver is inserted.
  2. You power off the line card in slot 4.
  3. While the line card is powered off, you remove the 4x100G-FR-S optic from port 8 and insert another optics other than the 4x100G-FR-S.
  4. You power on the line card in slot 4. Port Eth4/8 will not come up even after connecting it with any compatible port and transceiver combination on the peer end.

Configuring Breakout Ports With Profiles and Selectors Using the GUI

This procedure uses port profiles and selectors to configure breakout ports. You configure a breakout leaf port with an leaf interface profile, associate the profile with a switch, and configure the sub-ports.

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, the Cisco Application Policy Infrastructure Controllers (APICs) are online, and the Cisco APIC cluster is formed and healthy.
- A Cisco APIC fabric administrator account is available that can configure the breakout ports.
- The target leaf switches are registered in the Cisco ACI fabric and available.
- The 40GE or 100GE leaf switch ports are connected with Cisco breakout cables to the downlink ports.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the Navigation pane, expand **Interfaces** and **Leaf Interfaces** and **Profiles**.

Step 3 Right-click **Profiles** and choose **Create Leaf Interface Profile**.

Step 4 Type the name and optional description, click the + symbol on **Interface Selectors**

Step 5 Perform the following:

a) Type a name (and optional description) for the **Access Port Selector**.
b) In the **Interface IDs** field, type the slot and port for the breakout port.
c) In the **Interface Policy Group** field, click the down arrow and choose **Create Leaf Breakout Port Group**.
d) Type the name (and optional description) for the **Leaf Breakout Port Group**.
e) In the **Breakout Map** field, choose **10g-4x** or **25g-4x**.

For a list of the switches that support breakout, see Configuration of Breakout Ports, on page 193.

f) Click **Submit**.

Step 6 To assign a Breakout Port to an EPG, perform the following steps:

On the menu bar, choose **Tenant** > **Application Profiles** > **Application EPG**. Right-click on **Application EPG** to open **Create Application EPG**dialog box, and perform the following steps:

a) Select the **Statically Link with Leaves/Paths** check box to gain access to the **Leaves/Paths** tab in the dialog box.
b) Complete one of the following sets of steps:

Option	Description
If you want to deploy the EPG on...	Then
A node	**a.** Expand the **Leaves** area. **b.** From the **Node** drop-down list, choose a node. **c.** In the **Encap** field, enter the appropriate VLAN. **d.** (Optional) From the **Deployment Immediacy** drop-down list, accept the default **On Demand** or choose **Immediate**. **e.** (Optional) From the Mode drop-down list, accept the default **Trunk** or choose another mode.
A port on the node	**a.** Expand the **Paths** area.

Option	Description
	b. From the **Path** drop-down list, choose the appropriate node and port. **c.** (Optional) In the **Deployment Immediacy** field drop-down list, accept the default **On Demand** or choose **Immediate**. **d.** (Optional) From the Mode drop-down list, accept the default **Trunk** or choose another mode. **e.** In the **Port Encap** field, enter the secondary VLAN to be deployed. **f.** (Optional) In the **Primary Encap** field, enter the primary VLAN to be deployed.

Step 7 To associate the Leaf Interface Profile to a the leaf switch, perform the following steps:

a) Expand **Switches** and **Leaf Switches**, and **Profiles**.
b) Right-click **Profiles** and select **Create Leaf Profiles**.
c) Type the name and optional description of the Leaf Profile.
d) Click the **+** symbol on the **Leaf Selectors** area.
e) Type the leaf selector name and an optional description.
f) Click the down arrow on the **Blocks** field and choose the switch to be associated with the breakout leaf interface profile.
g) Click the down arrow on the **Policy Group** field and choose **Create Access Switch Policy Group**.
h) Type a name and optional description for the Access Switch Policy Group.
i) Optional. Enable other policies.
j) Click **Submit**.
k) Click **Update**.
l) Click **Next**.
m) In the **Associations Interface Selector Profiles** area, choose the Interface Selector Profile you previously created for the breakout port.
n) Click **Finish**.

Step 8 To verify the breakout port has been split into four sub ports, perform the following steps:

a) On the Menu bar, click **Fabric** > **Inventory**.
b) On the Navigation bar, click the Pod and Leaf where the breakout port is located.
c) Expand **Interfaces** and **Physical Interfaces**.
You should see four ports at the position where the breakout port was configured. For example, if you configured 1/10 as a breakout port, you should see the following:

- **eth1/10/1**
- **eth1/10/2**
- **eth1/10/3**
- **eth1/10/4**

Step 9 To configure the sub ports, perform the following steps:

a) On the Menu bar, click **Fabric** > **Access Policies**.
b) On the Navigation bar, expand **Interfaces**, **Leaf Interfaces**, **Profiles**, and the breakout leaf interface profile you previously created.

You will see a port selector with the breakout cable. Instead of defining a sub port block under already existing port selector, you need to define on a new access port selector.

c) On the Navigtion bar, right click the higher level interface profile and select **Create Access Port Selector**.
d) In the **Name** field, enter the sub port name.
e) In the **Interface IDs** field, enter the IDs for the four sub ports in a format such as `1/10/1-4`.
f) In the **Interface Policy Group** field, select **Create Leaf Access Port Policy Group**.
g) Click **Submit**.

Step 10 To apply the Policy Group to an individual interface that links the AAEP to the port, perform the following steps:

a) In the **Name** field, enter the name for the Leaf Access Port Group Policy.
b) In the **Link Level Policy** field, select **link-level_auto**.
c) In the **CDP Policy** field, select **cdp_enabled**.
d) In the **LLDP Policy** field, select **default**.
e) In the **Attached Entity Profile** field, select the AAEP profile to attach to the policy group.
f) Click **Submit**.

Configuring Breakout Ports With Profiles and Selectors Using the GUI

This procedure uses port profiles and selectors to configure breakout ports. You configure a breakout leaf port with an leaf interface profile, associate the profile with a switch, and configure the sub-ports.

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, the Cisco Application Policy Infrastructure Controllers (APICs) are online, and the Cisco APIC cluster is formed and healthy.
- A Cisco APIC fabric administrator account is available that can configure the breakout ports.
- The target leaf switches are registered in the Cisco ACI fabric and available.
- The 40GE or 100GE leaf switch ports are connected with Cisco breakout cables to the downlink ports.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.
Step 2 In the Navigation pane, expand **Interfaces** and **Leaf Interfaces** and **Profiles**.
Step 3 Right-click **Profiles** and choose **Create Leaf Interface Profile**.
Step 4 Type the name and optional description, click the + symbol on **Interface Selectors**
Step 5 Perform the following:

a) Type a name (and optional description) for the **Access Port Selector**.
b) In the **Interface IDs** field, type the slot and port for the breakout port.
c) In the **Interface Policy Group** field, click the down arrow and choose **Create Leaf Breakout Port Group**.

d) Type the name (and optional description) for the **Leaf Breakout Port Group**.
e) In the **Breakout Map** field, choose **10g-4x** or **25g-4x**.

For a list of the switches that support breakout, see Configuration of Breakout Ports, on page 193.

f) Click **Submit**.

Step 6 To assign a Breakout Port to an EPG, perform the following steps:

On the menu bar, choose **Tenant** > **Application Profiles** > **Application EPG**. Right-click on **Application EPG** to open **Create Application EPG**dialog box, and perform the following steps:

a) Select the **Statically Link with Leaves/Paths** check box to gain access to the **Leaves/Paths** tab in the dialog box.
b) Complete one of the following sets of steps:

Option	Description
If you want to deploy the EPG on...	Then
A node	**a.** Expand the **Leaves** area. **b.** From the **Node** drop-down list, choose a node. **c.** In the **Encap** field, enter the appropriate VLAN. **d.** (Optional) From the **Deployment Immediacy** drop-down list, accept the default **On Demand** or choose **Immediate**. **e.** (Optional) From the Mode drop-down list, accept the default **Trunk** or choose another mode.
A port on the node	**a.** Expand the **Paths** area. **b.** From the **Path** drop-down list, choose the appropriate node and port. **c.** (Optional) In the **Deployment Immediacy** field drop-down list, accept the default **On Demand** or choose **Immediate**. **d.** (Optional) From the Mode drop-down list, accept the default **Trunk** or choose another mode. **e.** In the **Port Encap** field, enter the secondary VLAN to be deployed. **f.** (Optional) In the **Primary Encap** field, enter the primary VLAN to be deployed.

Step 7 To associate the Leaf Interface Profile to a the leaf switch, perform the following steps:

a) Expand **Switches** and **Leaf Switches**, and **Profiles**.
b) Right-click **Profiles** and select **Create Leaf Profiles**.
c) Type the name and optional description of the Leaf Profile.
d) Click the + symbol on the **Leaf Selectors** area.
e) Type the leaf selector name and an optional description.
f) Click the down arrow on the **Blocks** field and choose the switch to be associated with the breakout leaf interface profile.
g) Click the down arrow on the **Policy Group** field and choose **Create Access Switch Policy Group**.
h) Type a name and optional description for the Access Switch Policy Group.

i) Optional. Enable other policies.
j) Click **Submit**.
k) Click **Update**.
l) Click **Next**.
m) In the **Associations Interface Selector Profiles** area, choose the Interface Selector Profile you previously created for the breakout port.
n) Click **Finish**.

Step 8 To verify the breakout port has been split into four sub ports, perform the following steps:

a) On the Menu bar, click **Fabric** > **Inventory**.
b) On the Navigation bar, click the Pod and Leaf where the breakout port is located.
c) Expand **Interfaces** and **Physical Interfaces**.
You should see four ports at the position where the breakout port was configured. For example, if you configured 1/10 as a breakout port, you should see the following:

- **eth1/10/1**
- **eth1/10/2**
- **eth1/10/3**
- **eth1/10/4**

Step 9 To configure the sub ports, perform the following steps:

a) On the Menu bar, click **Fabric** > **Access Policies**.
b) On the Navigation bar, expand **Interfaces**, **Leaf Interfaces**, **Profiles**, and the breakout leaf interface profile you previously created.

You will see a port selector with the breakout cable. Instead of defining a sub port block under already existing port selector, you need to define on a new access port selector.

c) On the Navigtion bar, right click the higher level interface profile and select **Create Access Port Selector**.
d) In the **Name** field, enter the sub port name.
e) In the **Interface IDs** field, enter the IDs for the four sub ports in a format such as `1/10/1-4`.
f) In the **Interface Policy Group** field, select **Create Leaf Access Port Policy Group**.
g) Click **Submit**.

Step 10 To apply the Policy Group to an individual interface that links the AAEP to the port, perform the following steps:

a) In the **Name** field, enter the name for the Leaf Access Port Group Policy.
b) In the **Link Level Policy** field, select **link-level_auto**.
c) In the **CDP Policy** field, select **cdp_enabled**.
d) In the **LLDP Policy** field, select **default**.
e) In the **Attached Entity Profile** field, select the AAEP profile to attach to the policy group.
f) Click **Submit**.

Configuring Breakout Ports With the Interface Configuration Using the GUI

Beginning with the Cisco Application Policy Infrastructure Controller (APIC) 6.0(1) release, you can use the **Fabric** > **Access Policies** > **Interface Configuration** workflow to configure breakout ports.

Before you begin

- The Cisco Application Centric Infrastructure (ACI) fabric is installed, the Cisco APICs are online, and the Cisco APIC cluster is formed and healthy.
- A Cisco APIC fabric administrator account is available that can configure the breakout ports.
- The target leaf switches are registered in the Cisco ACI fabric and available.
- The 40GE or 100GE leaf switch ports are connected with Cisco breakout cables to the downlink ports.

Procedure

Step 1 On the menu bar, choose **Fabric** > **Access Policies**.

Step 2 In the Navigation pane, choose **Interface Configuration**.

Step 3 In the Work pane, choose **Actions** > **Breakout**.

Step 4 In the **Breakout** page, perform the following substeps:

a) For **Node**, click **Select Node**, put a check in the box for the desired switch (node), then click **OK**. You can select multiple switches.
b) For **Interfaces For All Switches**, enter the range of desired interfaces.
c) For **Breakout Map**, choose the desired breakout type.
d) Click **Save**.

Configuring Dynamic Breakout Ports Using the NX-OS Style CLI

Use the following steps to configure a breakout port, verify the configuration, and configure an EPG on a sub port, using the NX-OS style CLI.

Before you begin

- The ACI fabric is installed, APIC controllers are online, and the APIC cluster is formed and healthy.
- An APIC fabric administrator account is available that will enable creating the necessary fabric infrastructure configurations.
- The target leaf switches are registered in the ACI fabric and available.
- The 40GE or 100GE leaf switch ports are connected with Cisco breakout cables to the downlink ports.

SUMMARY STEPS

1. **configure**
2. **leaf** *ID*
3. **interface ethernet***slot/port*
4. **breakout10g-4x** | **25g-4x**
5. **show run**
6. **tenant** *tenant-name*
7. **vrf context** *vrf-name*
8. **bridge-domain** *bridge-domain-name*
9. **vrf member** *vrf-name*
10. **application** *application-profile-name*
11. **epg** *epg-name*
12. **bridge-domain member** *bridge-domain-name*
13. **leaf** *leaf-name*
14. **speed** *interface-speed*
15. **show run**

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	**configure** **Example:** `apic1# configure`	Enters configuration mode.
Step 2	**leaf** *ID* **Example:** `apic1(config)# leaf 101`	Selects the leaf switch where the breakout port will be located and enters leaf configuration mode.
Step 3	**interface ethernet***slot/port* **Example:** `apic1(config-leaf)# interface ethernet 1/16`	Identifies the interface to be enabled as a 40 Gigabit Ethernet (GE) breakout port.
Step 4	**breakout10g-4x**	**25g-4x** **Example:** `apic1(config-leaf-if)# breakout 10g-4x`	Enables the selected interface for breakout. **Note** For switch support for the Dynamic Breakout Port feature, see Configuration of Breakout Ports, on page 193.

Step 5	**show run**
	Example:
	`apic1(config-leaf-if)# show run`
	`# Command: show running-config leaf 101 interface ethernet 1 / 16`
	`# Time: Fri Dec  2 18:13:39 2016`
		Verifies the configuration by showing the running configuration of the interface and returns to global configuration mode.

	Command or Action	Purpose

	`leaf 101`
	`interface ethernet 1/16`
	`breakout 10g-4x`
	`apic1(config-leaf-if)# exit`
	`apic1(config-leaf)# exit`	

Step 6	**tenant** *tenant-name* **Example:** `apic1(config)# tenant tenant64`	Selects or creates the tenant that will consume the breakout ports and enters tenant configuration mode.

Step 7	**vrf context** *vrf-name*
	Example:
	`apic1(config-tenant)# vrf context vrf64`
	`apic1(config-tenant-vrf)# exit`
		Creates or identifies the Virtual Routing and Forwarding (VRF) instance associated with the tenant and exits the configuration mode.

Step 8	**bridge-domain** *bridge-domain-name* **Example:** `apic1(config-tenant)# bridge-domain bd64`	Creates or identifies the bridge-domain associated with the tenant and enters BD configuration mode.

Step 9	**vrf member** *vrf-name*
	Example:
	`apic1(config-tenant-bd)# vrf member vrf64`
	`apic1(config-tenant-bd)# exit`
		Associates the VRF with the bridge-domain and exits the configuration mode.

Step 10	**application** *application-profile-name* **Example:** `apic1(config-tenant)# application app64`	Creates or identifies the application profile associated with the tenant and the EPG.
Step 11	**epg** *epg-name* **Example:** `apic1(config-tenant)# epg epg64`	Creates or identifies the EPG and enters into EPG configuration mode.

Step 12	**bridge-domain member** *bridge-domain-name*
	Example:
	`apic1(config-tenant-app-epg)# bridge-domain member bd64`
	`apic1(config-tenant-app-epg)# exit`
	`apic1(config-tenant-app)# exit`
	`apic1(config-tenant)# exit`
		Associates the EPG with the bridge domain and returns to global configuration mode.
		Configure the sub ports as desired, for example, use the speed command in leaf interface mode to configure a sub port.

Step 13	**leaf** *leaf-name*
	Example:
	`apic1(config)# leaf 1017`
	`apic1(config-leaf)# interface ethernet 1/13`
	`apic1(config-leaf-if)# vlan-domain member dom1`
	`apic1(config-leaf-if)# switchport trunk allowed vlan 20 tenant t1 application AP1 epg EPG1`
	Note
		Associates the EPG with a break-out port.

	Command or Action	Purpose
	The vlan-domain and vlan-domain member commands mentioned in the above example are a pre-requisite for deploying an EPG on a port.	

Step 14	**speed** *interface-speed*
	Example:
	`apic1(config)# leaf 101`
	`apic1(config-leaf)# interface ethernet 1/16/1`
	`apic1(config-leaf-if)# speed 10G`
	`apic1(config-leaf-if)# exit`
		Enters leaf interface mode, sets the speed of an interface, and exits the configuration mode.

Step 15	**show run** **Example:** `apic1(config-leaf)# show run`	After you have configured the sub ports, entering this command in leaf configuration mode displays the sub port details.

The port on leaf 101 at interface 1/16 is confirmed enabled for breakout with sub ports 1/16/1, 1/16/2, 1/16/3, and 1/16/4.

Example

This example configures the port for breakout:

```
apic1# configure
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/16
apic1(config-leaf-if)# breakout 10g-4x
```

This example configures the EPG for the sub ports.

```
apic1(config)# tenant tenant64
apic1(config-tenant)# vrf context vrf64
apic1(config-tenant-vrf)# exit
apic1(config-tenant)# bridge-domain bd64
apic1(config-tenant-bd)# vrf member vrf64
apic1(config-tenant-bd)# exit
apic1(config-tenant)# application app64
apic1(config-tenant-app)# epg epg64
apic1(config-tenant-app-epg)# bridge-domain member bd64
apic1(config-tenant-app-epg)# end
```

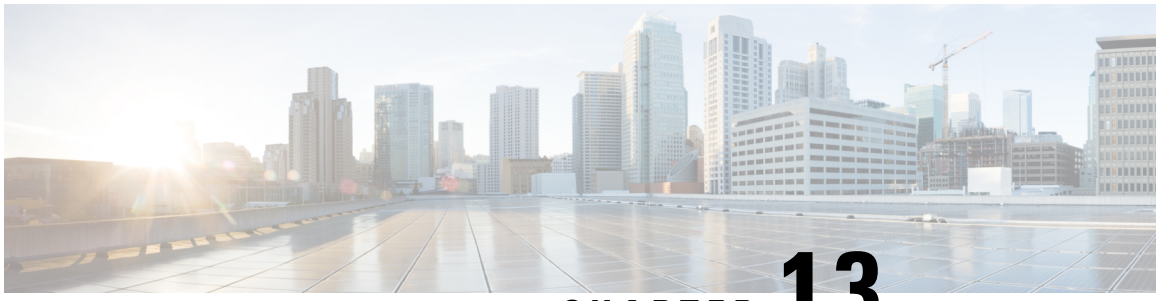
This example sets the speed for the breakout sub ports to 10G.

```
apic1(config)# leaf 101
apic1(config-leaf)# interface ethernet 1/16/1
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit

apic1(config-leaf)# interface ethernet 1/16/2
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/16/3
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit
apic1(config-leaf)# interface ethernet 1/16/4
apic1(config-leaf-if)# speed 10G
apic1(config-leaf-if)# exit
```

This example shows the four sub ports connected to leaf 101, interface 1/16.

```
apic1#(config-leaf)# show run
# Command: show running-config leaf 101
# Time: Fri Dec  2 00:51:08 2016
  leaf 101
    interface ethernet 1/16/1
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16/2
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16/3
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16/4
      speed 10G
      negotiate auto
      link debounce time 100
      exit
    interface ethernet 1/16
      breakout 10g-4x
      exit
    interface vfc 1/16
```



CHAPTER 13

Proxy ARP

This chapter contains the following sections:

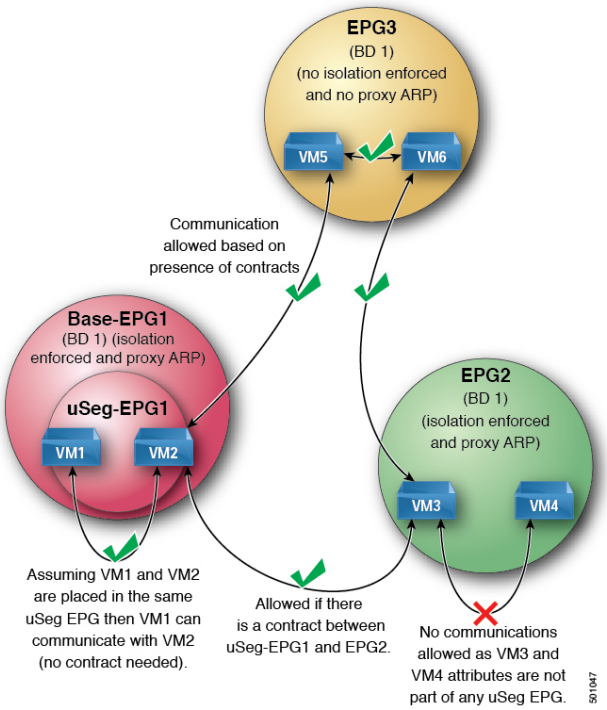
- About Proxy ARP, on page 211
- Guidelines and Limitations, on page 217
- Proxy ARP Supported Combinations, on page 217
- Configure Proxy ARP Using the Advanced GUI, on page 218
- Configure Proxy ARP Using the Cisco NX-OS Style CLI, on page 218

About Proxy ARP

Proxy ARP in Cisco ACI enables endpoints within a network or subnet to communicate with other endpoints without knowing the real MAC address of the endpoints. Proxy ARP is aware of the location of the traffic destination, and offers its own MAC address as the final destination instead.

To enable Proxy ARP, intra-EPG endpoint isolation must be enabled on the EPG see the following figure for details. For more information about intra-EPG isolation and Cisco ACI, see the *Cisco ACI Virtualization Guide*.

Figure 34: Proxy ARP and Cisco APIC



Proxy ARP within the Cisco ACI fabric is different from the traditional proxy ARP. As an example of the communication process, when proxy ARP is enabled on an EPG, if an endpoint A sends an ARP request for endpoint B and if endpoint B is learned within the fabric, then endpoint A will receive a proxy ARP response from the bridge domain (BD) MAC. If endpoint A sends an ARP request for endpoint B, and if endpoint B is not learned within the ACI fabric already, then the fabric will send a proxy ARP request within the BD. Endpoint B will respond to this proxy ARP request back to the fabric. At this point, the fabric does not send a proxy ARP response to endpoint A, but endpoint B is learned within the fabric. If endpoint A sends another ARP request to endpoint B, then the fabric will send a proxy ARP response from the BD MAC.

The following example describes the proxy ARP resolution steps for communication between clients VM1 and VM2:

1. VM1 to VM2 communication is desired.

Figure 35: VM1 to VM2 Communication is Desired.

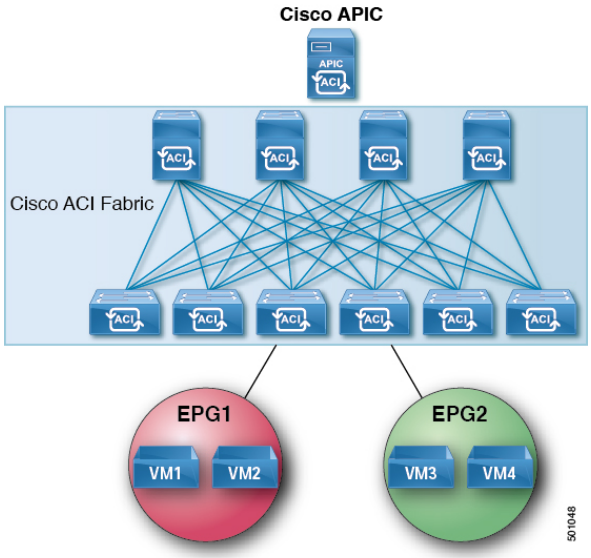


Table 8: ARP Table State

Device	State
VM1	IP = * MAC = *
ACI fabric	IP = * MAC = *
VM2	IP = * MAC = *

2. VM1 sends an ARP request with a broadcast MAC address to VM2.

Figure 36: VM1 sends an ARP Request with a Broadcast MAC address to VM2

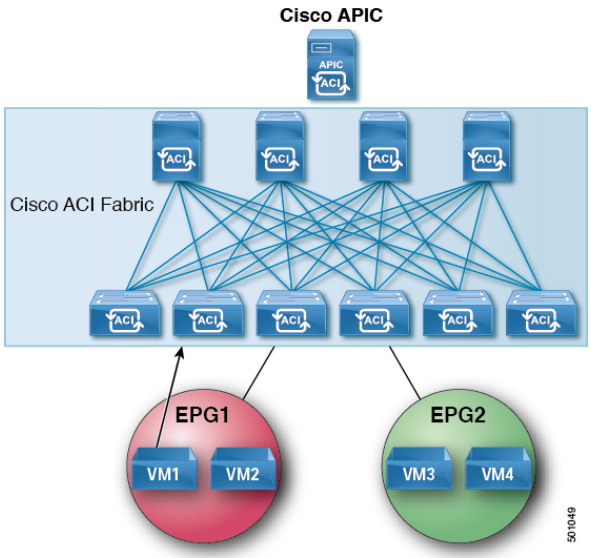


Table 9: ARP Table State

Device	State
VM1	IP = VM2 IP; MAC = ?
ACI fabric	IP = VM1 IP; MAC = VM1 MAC
VM2	IP = * MAC = *

3. The ACI fabric floods the proxy ARP request within the bridge domain (BD).

Figure 37: ACI Fabric Floods the Proxy ARP Request within the BD

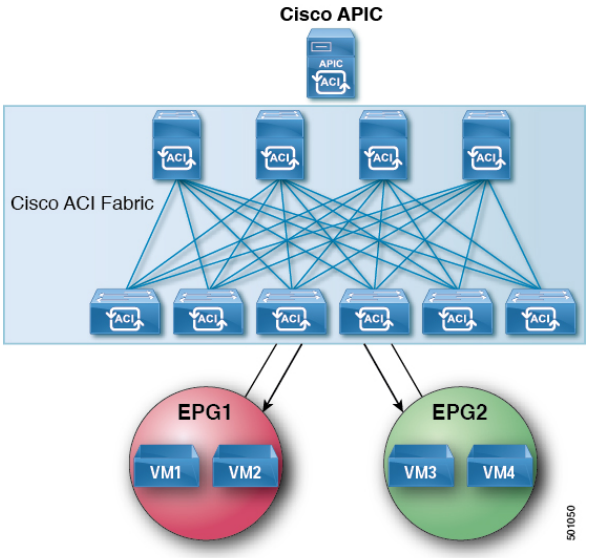


Table 10: ARP Table State

Device	State
VM1	IP = VM2 IP; MAC = ?
ACI fabric	IP = VM1 IP; MAC = VM1 MAC
VM2	IP = VM1 IP; MAC = BD MAC

4. VM2 sends an ARP response to the ACI fabric.

Figure 38: VM2 Sends an ARP Response to the ACI Fabric

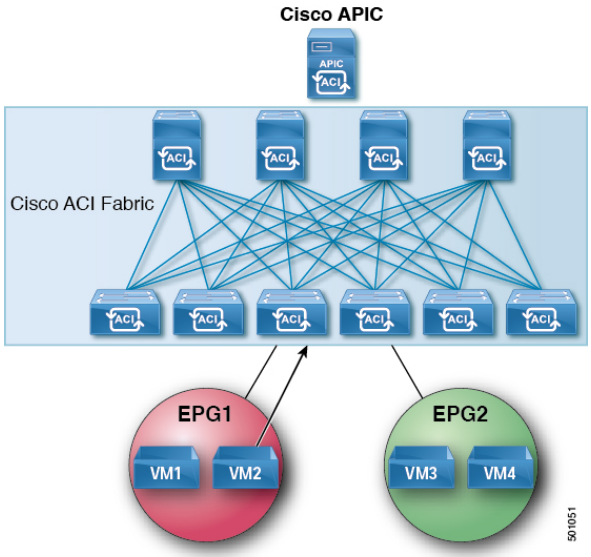


Table 11: ARP Table State

Device	State
VM1	IP = VM2 IP; MAC = ?
ACI fabric	IP = VM1 IP; MAC = VM1 MAC
VM2	IP = VM1 IP; MAC = BD MAC

5. VM2 is learned.

Figure 39: VM2 is Learned

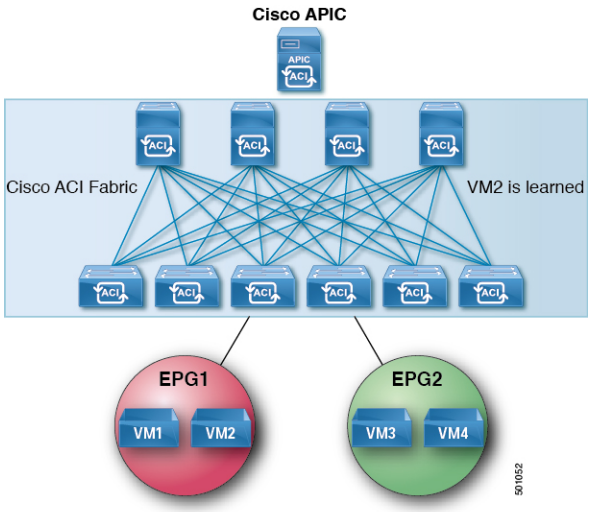


Table 12: ARP Table State

Device	State
VM1	IP = VM2 IP; MAC = ?
ACI fabric	IP = VM1 IP; MAC = VM1 MAC IP = VM2 IP; MAC = VM2 MAC
VM2	IP = VM1 IP; MAC = BD MAC

6. VM1 sends an ARP request with a broadcast MAC address to VM2.

Figure 40: VM1 Sends an ARP Request with a Broadcast MAC Address to VM2

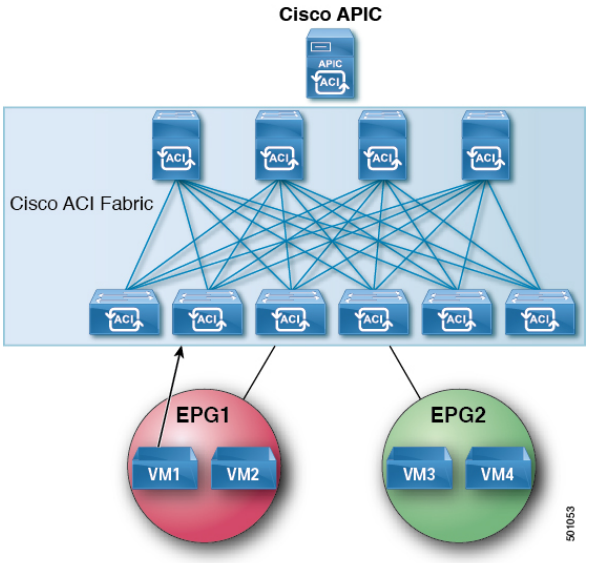


Table 13: ARP Table State

Device	State
VM1	IP = VM2 IP MAC = ?
ACI fabric	IP = VM1 IP; MAC = VM1 MAC IP = VM2 IP; MAC = VM2 MAC
VM2	IP = VM1 IP; MAC = BD MAC

7. The ACI fabric sends a proxy ARP response to VM1.

Figure 41: ACI Fabric Sends a Proxy ARP Response to VM1

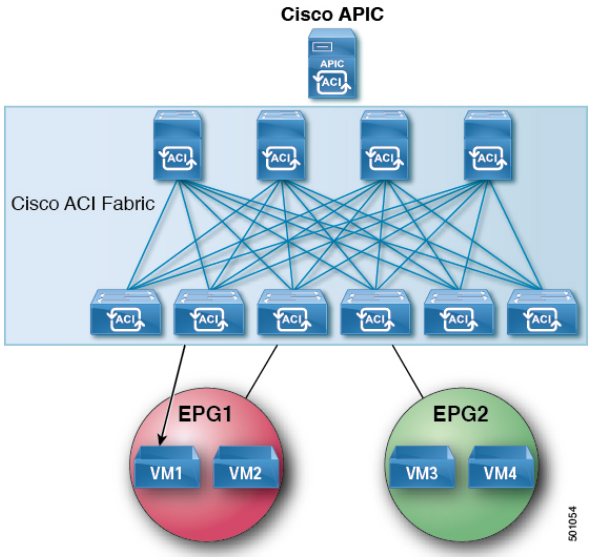


Table 14: ARP Table State

Device	State
VM1	IP = VM2 IP; MAC = BD MAC
ACI fabric	IP = VM1 IP; MAC = VM1 MAC IP = VM2 IP; MAC = VM2 MAC
VM2	IP = VM1 IP; MAC = BD MAC

Guidelines and Limitations

- Proxy ARP is supported only on isolated EPGs. If an EPG is not isolated, a fault will be raised.
- For communication within isolated EPGs with proxy ARP enabled, you must configure uSeg EPGs that match the IP address range of the VMs.
- ARP requests between isolated and regular endpoints do not use proxy ARP; endpoints communicate using the real MAC addresses of destination VMs.

Proxy ARP Supported Combinations

The following proxy ARP table provides the supported combinations between Regular EPG and Isolated Enforced EPG with Proxy ARP.

Table 15: Proxy ARP Supported Combinations

ARP From/To	Regular EPG	Isolated Enforced EPG with Proxy ARP
Regular EPG	ARP	ARP
Isolated Enforced EPG with Proxy ARP	ARP	Proxy ARP

Configure Proxy ARP Using the Advanced GUI

Before you begin

- The appropriate tenant, VRF, bridge domain, application profile and EPG must be created.
- Intra-EPG isolation must be enabled on the EPG where proxy ARP has to be enabled.

Procedure

Step 1 On the menu bar, click **Tenant** > **Tenant_name** .

Step 2 In the **Navigation** pane, expand the **Tenant_name** > **Application Profiles** > **Application_Profile_name** > **Application EPGs** , right click **Create Application EPG** dialog box to perform the following actions in the **Create Application EPG** dialog box:

a) In the **Name** field, add an EPG name.

Step 3 In the **Intra EPG Isolation** field, choose **Enforced** .
When Intra EPG isolation is enforced, the **Forwarding Control** field becomes available.

Step 4 In the **Forwarding Control** field, check the check box for **proxy-arp** .
This enables proxy-arp.

Step 5 In the **Bridge Domain** field, choose the appropriate bridge domain to associate from the drop-down list.

Step 6 Choose the remaining fields in the dialog box as appropriate, and click **Finish** .

Configure Proxy ARP Using the Cisco NX-OS Style CLI

Before you begin

- The appropriate tenant, VRF, bridge domain, application profile and EPG must be created.
- Intra-EPG isolation must be enabled on the EPG where proxy ARP has to be enabled.

Procedure

Step 1 **configure**

Enters configuration mode.

Example:

```
apic1#
configure
```

Step 2 **tenant** *tenant-name*

Enters the tenant configuration mode.

Example:

```
apic1(config)#
tenant Tenant1
```

Step 3 **application** *application-profile-name*

Creates an application profile and enters the application mode.

Example:

```
apic1(config-tenant)#
application Tenant1-App
```

Step 4 **epg** *application-profile-EPG-name*

Creates an EPG and enter the EPG mode.

Example:

```
apic1(config-tenant-app)#
epg Tenant1-epg1
```

Step 5 **proxy-arp enable**

Enables proxy ARP.

Note
You can disable proxy-arp with the **no proxy-arp** command.

Example:

```
apic1(config-tenant-app-epg)#
proxy-arp enable
```

Step 6 **exit**

Returns to application profile mode.

Example:

```
apic1(config-tenant-app-epg)#
exit
```

Step 7 **exit**

Returns to tenant configuration mode.

Example:

```
apic1(config-tenant-app)#
exit
```

Step 8 **exit**

Returns to global configuration mode.

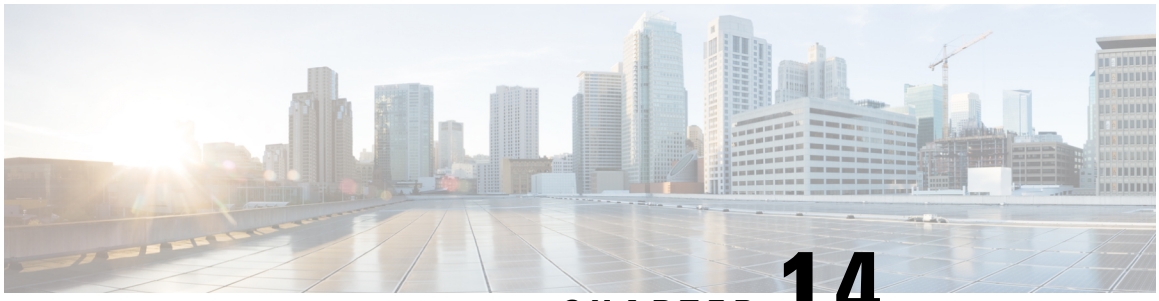
Example:

```
apic1(config-tenant)#
exit
```

Examples

This example shows how to configure proxy ARP.

```
apic1# conf t
apic1(config)# tenant Tenant1
apic1(config-tenant)# application Tenant1-App
apic1(config-tenant-app)# epg Tenant1-epg1
apic1(config-tenant-app-epg)# proxy-arp enable
apic1(config-tenant-app-epg)#
apic1(config-tenant)#
```



CHAPTER 14

Traffic Storm Control

This chapter contains the following sections:

- About Traffic Storm Control, on page 221
- Storm Control Guidelines and Limitations, on page 221
- Configuring a Traffic Storm Control Policy Using the GUI, on page 224
- Configuring a Traffic Storm Control Policy Using the NX-OS Style CLI, on page 225
- Configuring a Storm Control SNMP Trap, on page 227

About Traffic Storm Control

A traffic storm occurs when packets flood the LAN, creating excessive traffic and degrading network performance. You can use traffic storm control policies to prevent disruptions on Layer 2 ports by broadcast, unknown multicast, or unknown unicast traffic storms on physical interfaces.

By default, storm control is not enabled in the ACI fabric. ACI bridge domain (BD) Layer 2 unknown unicast flooding is enabled by default within the BD but can be disabled by an administrator. In that case, a storm control policy only applies to broadcast and unknown multicast traffic. If Layer 2 unknown unicast flooding is enabled in a BD, then a storm control policy applies to Layer 2 unknown unicast flooding in addition to broadcast and unknown multicast traffic.

Traffic storm control (also called traffic suppression) allows you to monitor the levels of incoming broadcast, multicast, and unknown unicast traffic over a one second interval. During this interval, the traffic level, which is expressed either as percentage of the total available bandwidth of the port or as the maximum packets per second allowed on the given port, is compared with the traffic storm control level that you configured. When the ingress traffic reaches the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends. An administrator can configure a monitoring policy to raise a fault when a storm control threshold is exceeded.

Storm Control Guidelines and Limitations

Configure traffic storm control levels according to the following guidelines and limitations:

- Typically, a fabric administrator configures storm control in fabric access policies on the following interfaces:
  - A regular trunk interface.

- A direct port channel on a single leaf switch.
- A virtual port channel (a port channel on two leaf switches).

- Beginning with release 4.2(1), support is now available for triggering SNMP traps from Cisco Application Centric Infrastructure (ACI) when storm control thresholds are met, with the following restrictions:
  - There are two actions associated with storm control: drop and shutdown. With the shutdown action, interface traps will be raised, but the storm control traps to indicate that the storm is active or clear is not determined by the shutdown action. Storm control traps with the shutdown action on the policy should therefore be ignored.
  - If the ports flap with the storm control policy on, clear and active traps are seen together when the stats are collected. Clear and active traps are typically not seen together, but this is expected behavior in this case.

- For port channels and virtual port channels, the storm control values (packets per second or percentage) apply to all individual members of the port channel.



Note For switch hardware, beginning with Cisco Application Policy Infrastructure Controller (APIC) release 1.3(1) and switch release 11.3(1), for port channel configurations, the traffic suppression on the aggregated port may be up to two times the configured value. The new hardware ports are internally subdivided into these two groups: slice-0 and slice-1. To check the slicing map, use the `vsh_lc` command `show platform internal hal l2 port gpd` and look for `slice 0` or `slice 1` under the `Sl` column. If port channel members fall on both slice-0 and slice-1, allowed storm control traffic may become twice the configured value because the formula is calculated based on each slice.

- When configuring by percentage of available bandwidth, a value of 100 means no traffic storm control and a value of 0.01 suppresses all traffic.
- Due to hardware limitations and the method by which packets of different sizes are counted, the level percentage is an approximation. Depending on the sizes of the frames that make up the incoming traffic, the actual enforced level might differ from the configured level by several percentage points. Packets-per-second (PPS) values are converted to percentage based on 256 bytes.
- Maximum burst is the maximum accumulation of rate that is allowed when no traffic passes. When traffic starts, all the traffic up to the accumulated rate is allowed in the first interval. In subsequent intervals, traffic is allowed only up to the configured rate. The maximum supported is 65535 KB. If the configured rate exceeds this value, it is capped at this value for both PPS and percentage.
- The maximum burst that can be accumulated is 512 MB.
- On an egress leaf switch in optimized multicast flooding (OMF) mode, traffic storm control will not be applied.
- On an egress leaf switch in non-OMF mode, traffic storm control will be applied.
- Storm control does not police spanning tree protocol (STP) and OSPF traffic.
- On a leaf switch for FEX, traffic storm control is not available on host-facing interfaces.

- Traffic storm control unicast/multicast differentiation is not supported on Cisco Nexus C93128TX, C9396PX, C9396TX, C93120TX, C9332PQ, C9372PX, C9372TX, C9372PX-E, or C9372TX-E switches.
- SNMP traps for traffic storm control are not supported on Cisco Nexus C93128TX, C9396PX, C9396TX, C93120TX, C9332PQ, C9372PX, C9372TX, C9372PX-E, C9372TX-E switches.
- Traffic storm control traps is not supported on Cisco Nexus C93128TX, C9396PX, C9396TX, C93120TX, C9332PQ, C9372PX, C9372TX, C9372PX-E, or C9372TX-E switches.
- Storm Control Action is supported only on physical Ethernet interfaces and port channel interfaces.

  Beginning with release 4.1(1), Storm Control **Shutdown** option is supported. When the **shutdown** action is selected for an interface with the default Soak Instance Count, the packets exceeding the threshold are dropped for 3 seconds and the port is shutdown on the 3rd second. The default action is **Drop**. When **Shutdown** action is selected, the user has the option to specify the soaking interval. The default soaking interval is 3 seconds. The configurable range is from 3 to 10 seconds.
- If the data plane policing (DPP) policer that is configured for the interface has a value that is lower than storm policer's value, the DPP policer will take the precedence. The lower value that is configured between the DPP policer and storm policer is honored on the configured interface.
- Beginning with release 4.2(6), the storm policer is enforced for all forwarded control traffic in the leaf switch for the DHCP, ARP, ND, HSRP, PIM, IGMP, and EIGRP protocols regardless of whether the bridge domain is configured for **Flood in BD** or **Flood in Encapsulation**. This behavior change applies only to EX and later leaf switches.
  - With EX switches, you can configure both the supervisor policer and storm policer for one of the protocols. In this case, if a server sends traffic at a rate higher than the configured supervisor policer rate (Control Plane Policing, CoPP), then the storm policer will allow more traffic than what is configured as the storm policer rate. If the incoming traffic rate is equal to or less than supervisor policer rate, then the storm policer will correctly allow the configured storm traffic rate. This behavior is applicable irrespective of the configured supervisor policer and storm policer rates.
  - One side effect of the storm policer now being enforced for all forwarded control traffic in the leaf switch for the specified protocols is that control traffic that gets forwarded in the leaf switch will now get subjected to storm policer drops. In previous releases, no such storm policer drops occur for the protocols that are affected by this behavior change.
- Traffic storm control cannot police multicast traffic in a bridge domain or VRF instance that has PIM enabled.
- When the storm control policer is applied on a port channel interface, the allowed rate may be more than the configured rate. If the member links of the port channel span across multiple slices, then the allowed traffic rate will be equal to the configured rate multiplied by the number of slices across which the member links span.

  The port-to-slice mapping depends on the switch model.

  As an example, assume that there is a port channel that has member links port1, port2, and port3 with a storm policer rate of 10Mbps.
  - If port1, port2, and port3 belong to slice1, then traffic is policed to 10Mbps.
  - If port1 and port2 belong to slice1 and port3 belongs to slice2, then traffic is policed to 20Mbps.
  - If port1 belongs to slice1, port2 belongs to slice2, and port3 belongs to slice3, then traffic is policed to 30Mbps.

Configuring a Traffic Storm Control Policy Using the GUI

Procedure

Step 1 In the menu bar, click **Fabric**.

Step 2 In the submenu bar, click **Access Policies**.

Step 3 In the **Navigation** pane, expand **Policies**.

Step 4 Expand **Interface**.

Step 5 Right-click **Storm Control** and choose **Create Storm Control Interface Policy**.

Step 6 In the **Create Storm Control Interface Policy** dialog box, enter a name for the policy in the **Name** field.

Step 7 In the **Configure Storm Control** field, click the radio button for either **All Types** or **Unicast, Broadcast, Multicast**.

Note
Selecting the **Unicast, Broadcast, Multicast** radio button allows you to configure Storm Control on each traffic type separately.

Step 8 In the **Specify Policy In** field, click the radio button for either **Percentage** or **Packets Per Second**.

Step 9 If you chose **Percentage**, perform the following steps:

a) In the **Rate** field, enter a traffic rate percentage.

Enter a number between 0 and 100 that specifies a percentage of the total available bandwidth of the port. When the ingress traffic is either equal to or greater than this level during a one second interval, traffic storm control drops traffic for the remainder of the interval. A value of 100 means no traffic storm control. A value of 0 suppresses all traffic.

b) In the **Max Burst Rate** field, enter a burst traffic rate percentage.

Enter a number between 0 and 100 that specifies a percentage of the total available bandwidth of the port. When the ingress traffic is equal to or greater than, traffic storm control begins to drop traffic.

Note
The **Max Burst Rate** should be greater than or equal to the value of **Rate**.

Step 10 If you chose **Packets Per Second**, perform the following steps:

a) In the **Rate** field, enter a traffic rate in packets per second.

During this interval, the traffic level, expressed as packets flowing per second through the port, is compared with the traffic storm control level that you configured. When the ingress traffic is equal to or greater than the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends.

b) In the **Max Burst Rate** field, enter a burst traffic rate in packets per second.

During this interval, the traffic level, expressed as packets flowing per second through the port, is compared with the burst traffic storm control level that you configured. When the ingress traffic is equal to or greater than the traffic storm control level that is configured on the port, traffic storm control drops the traffic until the interval ends.

Step 11 The policy action can be altered from the default by selecting shutdown in the **Storm Control Action** and adjusting the default in **Storm Control Soak Count** fields.

Note
When the **shutdown** action is selected for an interface with the default Soak Instance Count, the packets exceeding the threshold are dropped for 3 seconds and the port is shutdown on the 3rd second.

Step 12 Click **Submit**.

Step 13 Apply the storm control interface policy to an interface port.

a) In the menu bar, click **Fabric**.
b) In the submenu bar, click **Access Policies**.
c) In the **Navigation** pane, expand **Interfaces**.
d) Expand **Leaf Interfaces**.
e) Expand **Policy Groups**.
f) Select **Leaf Policy Groups**.

Note
If your APIC version is earlier than 2.x, you select **Policy Groups**.

g) Select the leaf access port policy group, the PC interface policy group, the vPC interface policy group, or the PC/vPC override policy group to which you want to apply the storm control policy.
h) In the **Work** pane, click the drop down for **Storm Control Interface Policy** and select the created **Traffic Storm Control Policy**.
i) Click **Submit**.

Configuring a Traffic Storm Control Policy Using the NX-OS Style CLI

SUMMARY STEPS

1. Enter the following commands to create a PPS policy:
2. Enter the following commands to create a percent policy:
3. Configure storm control on physical ports, port channels, or virtual port channels:
4. To alter the policy action:
5. Configure the soak-instance count which is applicable for port **Shutdown** action only.

DETAILED STEPS

Procedure

	Command or Action	Purpose
Step 1	Enter the following commands to create a PPS policy: **Example:**	

	Command or Action	Purpose
	`(config)# template policy-group pg1` `(config-pol-grp-if)# storm-control pps 10000 burst-rate 10000`	

Step 2	Enter the following commands to create a percent policy:
	Example:
	`(config)# template policy-group pg2`
	`(config-pol-grp-if)# storm-control level 50 burst-rate 60`	

Step 3	Configure storm control on physical ports, port channels, or virtual port channels:
	Example:
	`[no] storm-control [unicast|multicast|broadcast] level <percentage> [burst-rate <percentage>]`
	`[no] storm-control [unicast|multicast|broadcast] pps <packet-per-second> [burst-rate <packet-per-second>]`
	
	`sd-tb2-ifc1# configure terminal`
	
	`sd-tb2-ifc1(config)# leaf 102`
	
	`sd-tb2-ifc1(config-leaf)# interface ethernet 1/19`
	
	`sd-tb2-ifc1(config-leaf-if)# storm-control unicast level 35 burst-rate 45`
	`sd-tb2-ifc1(config-leaf-if)# storm-control broadcast level 36 burst-rate 36`
	`sd-tb2-ifc1(config-leaf-if)# storm-control broadcast level 37 burst-rate 38`
	`sd-tb2-ifc1(config-leaf-if)#`
	
	`sd-tb2-ifc1# configure terminal`
	
	`sd-tb2-ifc1(config)# leaf 102`
	
	`sd-tb2-ifc1(config-leaf)# interface ethernet 1/19`
	
	`sd-tb2-ifc1(config-leaf-if)# storm-control broadcast pps 5000 burst-rate 6000`
	`sd-tb2-ifc1(config-leaf-if)# storm-control unicast pps 7000 burst-rate 7000`
	`sd-tb2-ifc1(config-leaf-if)# storm-control unicast pps 8000 burst-rate 10000`
	`sd-tb2-ifc1(config-leaf-if)#`	

Step 4	To alter the policy action:
	Example:
	`apic1(config-leaf-if)# storm-control action ?`
	`drop drop`
	`shutdown shutdown`	

Step 5	Configure the soak-instance count which is applicable for port **Shutdown** action only. **Example:**	

	Command or Action	Purpose
	apic-ifc1(config-leaf)# int eth 1/27 apic-ifc1(config-leaf-if)# storm-control soak-instance-count ? <3-10> Storm Control SI-Count Instances	

Configuring a Storm Control SNMP Trap

This section describes how to configure a storm control SNMP trap on leaf switches.

You can configure a storm control on SNMP trap using a trap name on the MIB definition. An event on MIB for an interface and when the storm is detected and cleared, a trap is filtered on the same leaf to configure the storm. You can configure the storm in two ways:

- Granular configuration—Sets the type of traffic such as, unicast, multicast and broadcast.
- Non-granular configuration—Sets all types of traffic.

For details on restrictions for triggering the SNMP traps from Cisco ACI when storm control thresholds are met, see Storm Control Guidelines and Limitations, on page 221. For details on Cisco Nexus switches that are not supported on traffic storm control traps, see the guidelines for Storm Control.

Storm Trap

The storm trap will be triggered whenever there is an event and the storm is active or cleared.

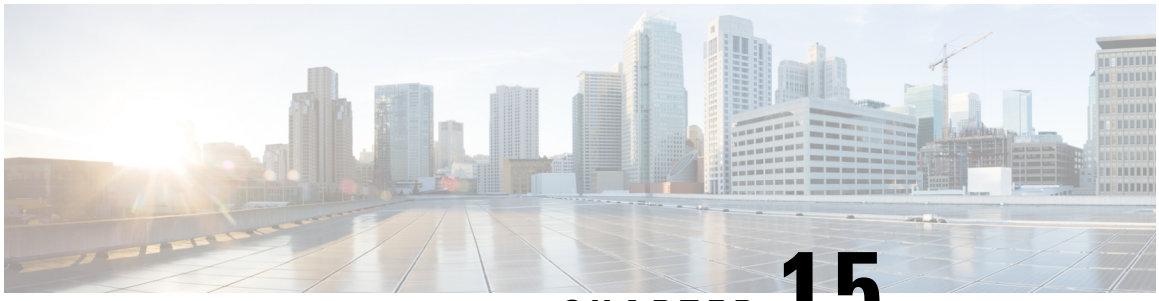
```
cpscEventRev1 NOTIFICATION-TYPE
        OBJECTS   { cpscStatus }
        STATUS     current
        DESCRIPTION
```

The implementation sends this notification when a storm event occurs on an interface with respect to a particular traffic type.

The storm status is updated in the fields: `bcDropIncreased`, `uucDropIncreased`, `mcDropIncreased`, and `dropIncreased` for broadcast, unicast, multicast and non-granular traffic types respectively of `dbgIfStorm` MO. The granular and non-granular configurations use flags to set the storm. When a strom is active the flag is set to 1 and when the storm is cleared the flag is set to 2. The following flags generate the events required for the SNMP trap trigger.

```
cat /mit/sys/phys-\[eth--1\]/dbgIfStorm/summary

# Interface Storm Drop Couters
bcDropBytes   :0
bcDropIncreased   :2
childAction    :
dn        :sys/phys-[eth/1]/dbgIfStorm
dropBytes      :0
dropIncreased  :2
mcDropBytes    :0
mcDropIncreased :2
modTs           :never
monPoIDn        :uni/infra/moninfra-default
m               :dbgIfStorm
status          :
uucDropBytes    :0
uucDropIncreased :2
```



CHAPTER 15

MACsec

This chapter contains the following sections:

- About MACsec, on page 229
- Guidelines and Limitations for MACsec on switches, on page 230
- Configuring MACsec for Fabric Links Using the GUI, on page 234
- Configuring MACsec for Access Links Using the GUI, on page 234
- Configuring MACsec Parameters Using the APIC GUI, on page 235
- Configuring MACsec Keychain Policy Using the GUI, on page 235
- Configuring MACsec Using the NX-OS Style CLI, on page 236

About MACsec

MACsec is an IEEE 802.1AE standards based Layer 2 hop-by-hop encryption that provides data confidentiality and integrity for media access independent protocols.

MACsec, provides MAC-layer encryption over wired networks by using out-of-band methods for encryption keying. The MACsec Key Agreement (MKA) Protocol provides the required session keys and manages the required encryption keys.

The 802.1AE encryption with MKA is supported on all types of links, that is, host facing links (links between network access devices and endpoint devices such as a PC or IP phone), or links connected to other switches or routers.

MACsec encrypts the entire data except for the Source and Destination MAC addresses of an Ethernet packet. The user also has the option to skip encryption up to 50 bytes after the source and destination MAC address.

To provide MACsec services over the WAN or Metro Ethernet, service providers offer Layer 2 transparent services such as E-Line or E-LAN using various transport layer protocols such as Ethernet over Multiprotocol Label Switching (EoMPLS) and L2TPv3.

The packet body in an EAP-over-LAN (EAPOL) Protocol Data Unit (PDU) is referred to as a MACsec Key Agreement PDU (MKPDU). When no MKPDU is received from a participants after 3 hearbeats (each hearbeat is of 2 seconds), peers are deleted from the live peer list. For example, if a client disconnects, the participant on the switch continues to operate MKA until 3 heartbeats have elapsed after the last MKPDU is received from the client.

APIC Fabric MACsec

The APIC will be responsible for the MACsec keychain distribution to all the nodes in a Pod or to particular ports on a node. Below are the supported MACsec keychain and MACsec policy distribution supported by the APIC.

- A single user provided keychain and policy per Pod
- User provided keychain and user provided policy per fabric interface
- Auto generated keychain and user provided policy per Pod

A node can have multiple policies deployed for more than one fabric link. When this happens, the per fabric interface keychain and policy are given preference on the affected interface. The auto generated keychain and associated MACsec policy are then given the least preference.

APIC MACsec supports two security modes. The MACsec **must secure** only allows encrypted traffic on the link while the **should secure** allows both clear and encrypted traffic on the link. Before deploying MACsec in **must secure** mode, the keychain must be deployed on the affected links or the links will go down. For example, a port can turn on MACsec in **must secure** mode before its peer has received its keychain resulting in the link going down. To address this issue the recommendation is to deploy MACsec in **should secure** mode and once all the links are up then change the security mode to **must secure**.

Note Any MACsec interface configuration change will result in packet drops.

MACsec policy definition consists of configuration specific to keychain definition and configuration related to feature functionality. The keychain definition and feature functionality definitions are placed in separate policies. Enabling MACsec per Pod or per interface involves deploying a combination of a keychain policy and MACsec functionality policy.

Note Using internal generated keychains do not require the user to specify a keychain.

APIC Access MACsec

MACsec encryption is configured under fabric policies for fabric ports and access policies for access ports and can be applied to physical interfaces, port-channels, or vPCs (virtual port-channels). Interfaces with MACsec enabled can operate as either Layer 2 interfaces (such as EPGs or L3Out SVIs) or Layer 3 interfaces (including L3Out routed interfaces or routed sub-interfaces). Enabling MACsec on an interface enables MACsec encryption on the entire interface and applies to all VLANs or sub-interfaces configured on the interface.

Guidelines and Limitations for MACsec on switches

MACsec is supported on the following switches:

- N9K-C93108TC-FX3P
- N9K-C93108TC-FX3 (beginning with the 6.0(5) release)

- N9K-C93108TC-FX
- N9K-C93180YC-FX3
- N9K-C93180YC-FX
- N9K-C93216TC-FX2
- N9K-C93240YC-FX2
- N9K-C9332C
- N9K-C93360YC-FX2
- N9K-C9336C-FX2
- N9K-C9348GC-FXP, only with 10G+
- N9K-C9364C
- N9K-C9364D-GX2A
- N9K-C9348D-GX2A
- N9K-C9332D-GX2B
- N9K-C9408 (beginning with the 6.0(2) release)

MACsec is supported on the following line cards:

- N9K-X9716D-GX
- N9K-X9736C-FX

Configure MACsec on switches according to the following guidelines and limitations:

- MACsec encryption is configured under fabric policies for fabric ports and access policies for access ports and can be applied to physical interfaces, port-channels, or vPCs (virtual port-channels). Interfaces with MACsec enabled can operate as either Layer 2 interfaces (such as EPGs or L3Out SVIs) or Layer 3 interfaces (including L3Out routed interfaces or routed sub-interfaces). Enabling MACsec on an interface enables MACsec encryption on the entire interface and applies to all VLANs or sub-interfaces configured on the interface.
- MACsec is not supported on 10G QSA modules.
- MACsec is not supported with 1G speed on a Cisco ACI leaf switch.
- MACsec between a Cisco ACI leaf switch and a computer host has not been validated by Cisco. Switch-to-switch mode has been validated and is supported.
- MACsec is supported on breakout ports.
- MACsec is supported on remote leaf switches.
- When using copper ports, the copper cables must be connected directly the peer device (standalone N9k) in 10G mode.
- A 10G copper SFP module on the peer is not supported.
- FEX ports are not supported for MACsec.

- The **must-secure** mode is not supported at the pod level.
- A MACsec policy with the name "default" is not supported.
- Auto-key generation is only supported at the pod level for fabric ports.
- Do not clean reboot a node if the fabric ports of that node is running MACsec in **must-secure** mode.
- Adding a new node to a pod or stateless reboot of a node in a pod that is running MACsec, **must-secure** mode requires changing the mode to **should-secure** for the node to join the pod.
- Only initiate an upgrade or downgrade if the fabric links are in the **should-secure** mode. After the upgrade or downgrade has completed, you can change the mode to **must-secure**. Upgrading or downgrading in the **must-secure** mode results in nodes losing connectivity to the fabric. Recovering from connectivity loss requires you to configure in **should-secure** mode the fabric links of the nodes that are visible to the Cisco Application Policy Infrastructure Controller (APIC). If the fabric was downgraded to a version which does not support MACsec, then nodes which are out of fabric will need to be clean rebooted.
- For a PC or vPC interface, MACsec can be deployed using policy groups per PC or vPC interface. Port selectors are used to deploy the policies to a particular set of ports.
- We recommend that you configure MACsec polices with the **should-secure** mode before you export a configuration.
- All of the links on a spine switch are considered to be fabric links. However, if a spine switch link is used for IPN connectivity, then this link will be treated as an access link. This means that a MACsec access policy must be used to deploy MACsec on these links.
- The 5.2(3) release added support for Multi-Pod Spines Back-to-Back. The following guidelines and limitations apply to MACsec with Multi-Pod Spines Back-to-Back:
  - The back-to-back links are treated as fabric links. You must create a MACsec fabric policy when enabling MACsec on the spine back-to-back links.

    For information about creating a MACsec fabric policy, see the "Configuring MACsec for Fabric Links Using the GUI" procedure in the *Cisco APIC Layer 2 Networking Configuration Guide*.
  - Both back-to-back links must use the same MACsec policy. If you used pod polices, then both pods should deploy the same MACsec policy.
- If a remote leaf fabric link is used for IPN connectivity, then this link will be treated as an access link. A MACsec access policy needs to be used to deploy MACsec on these links.
- Improper deployment of **must-secure** mode on remote leaf switch fabric links can result in loss of connectivity to the fabric. Follow the instructions provided in Deploying must-secure mode, on page 233 to prevent such issues.
- MACsec sessions can take up to a minute to form or tear down when a new key is added to an empty keychain or an active key is deleted from a keychain.
- Before reloading a line card or fabric module on a spine switch, all **must-secure** links should be changed to the **should-secure** mode. After the reload completes and the session comes up in the **should-secure** mode, change the mode to **must-secure**.
- When selecting the cipher suite AES 128 or AES 256 without Extended Packet Numbering (XPN), you must explicitly specify the Security Association Key (SAK) expiry time. Leaving the SAK expiry time value at the default ("disabled") can cause interfaces to go out of service randomly.

- A replay window is necessary to support the use of MACsec over provider networks that reorder frames. Frames within the window can be received out of order, but are not replay protected. The default window size is 64. The replay window size can be configured in the range of 0 to 2^{32}-1 if you use the Cisco APIC GUI or CLI. If you use a XPN cipher suite, the maximum replay window size is 2^{30}-1, and if you configure a higher window size, the window size gets restricted to 2^{30}-1. If you change the cipher suite to a non-XPN cipher suite, then there is no restriction and the configured window size is used.
- In the 5.2(2) release and later, if you are using MACsec on Cisco N9K-X9716D-GX line card fabric ports and then downgrade to a release prior to 5.2(2), the earlier releases do not support MACsec on this line card. However, no fault is raised to indicate that MACsec is not supported. In this scenario, if the peer leaf switch supports MACsec, the session comes up as secure. But, on the spine side, the session shows as pending.
- Link-level flow control (LLFC) and priority flow control (PFC) are not supported with MACsec.
- Cisco APIC does not support passing MACsec through its infrastrucutre for clients.

Deploying must-secure mode

Incorrectly deploying a policy that is configured for **must-secure** mode can result in a loss of connectivity. The procedure below should be followed in order to prevent such issues:

- You must ensure that each link pair has their keychains before enabling MACsec **must-secure** mode. To ensure this, we recommend that you deploy the policy in **should-secure** mode, and once MACsec sessions are active on the expected links, change the mode to **must-secure**.
- Attempting to replace the keychain on a MACsec policy that is configured to **must-secure** can cause links to go down. In this case, use the following procedure:
  1. Change MACsec policy that is using the new keychain to **should-secure** mode.
  2. Verify that the affected interfaces are using **should-secure** mode.
  3. Update MACsec policy to use new keychain.
  4. Verify that relevant interfaces with active MACsec sessions are using the new keychain.
  5. Change MACsec policy to **must-secure** mode.
- Use the following procedure to disable/remove a MACsec policy deployed in **must-secure** mode:
  1. Change the MACsec policy to **should-secure**.
  2. Verify that the affected interfaces are using **should-secure** mode.
  3. Disable/remove the MACsec policy.

Keychain Definition

- There should be one key in the keychain with a start time of **now**. If **must-secure** is deployed with a keychain that does not have a key that is immediately active then traffic will be blocked on that link until the key becomes current and a MACsec session is started. If **should-secure** mode is being used then traffic will be unencrypted until the key becomes current and a MACsec session has started.

- There should be one key in the keychain with an end time of **infinite**. When a keychain expires, then traffic is blocked on affected interfaces which are configured for **must-secure** mode. Interfaces configured for **should-secure** mode transmit unencrypted traffic.
- There should be overlaps in the end time and start time of keys that are used sequentially to ensure the MACsec session stays up when there is a transition between keys.

Configuring MACsec for Fabric Links Using the GUI

Procedure

Step 1 On the menu bar, click **Fabric** > **Fabric Policies** > **Policies** > **MACsec** > **Interfaces**. In the **Navigation** pane, right click on **Interfaces** to open **Create MACsec Fabric Interface Policy** and perform the following actions:

a) In the **Name** field, enter a name for the MACsec Fabric Interface policy.
b) In the **MACsec Parameters** field, either select a previously configured MACsec Parameters policy or create a new one.
c) In the **MACsec Keychain Policy** field, either select a previously configured MACsec Parameters policy or create a new one and click **Submit**.

To create a **MACsec Keychain Policy**, see Configuring MACsec Keychain Policy Using the GUI, on page 235.

Step 2 To apply the **MACsec Fabric Interface Policy** to a Fabric Leaf or Spine Port Policy Group, in the Navigation pane, click **Interfaces** > ***Leaf/Spine*** **Interfaces** > **Policy Groups** > ***Spine/Leaf*** **Port Policy Group_*name***. In the **Work** pane, select the **MACsec Fabric Interface Policy** just created.

Step 3 To apply the **MACsec Fabric Interface Policy** to a Pod Policy Group, in the Navigation pane, click **Pods** > **Policy Groups** > **Pod Policy Group_*name***. In the **Work** pane, select the **MACsec Fabric Interface Policy** just created.

Configuring MACsec for Access Links Using the GUI

Procedure

Step 1 On the menu bar, click **Fabric** > **External Access Policies**. In the **Navigation** pane, click on **Policies** > **Interface** > **MACsec** > **Interfaces** and right click on **Interfaces** to open **Create MACsec Fabric Interface Policy** and perform the following actions:

a) In the **Name** field, enter a name for the MACsec Access Interface policy.
b) In the **MACsec Parameters** field, either select a previously configured MACsec Parameters policy or create a new one.
c) In the **MACsec Keychain Policy** field, either select a previously configured MACsec Parameters policy or create a new one and click **Submit**.

To create a **MACsec Keychain Policy**, see Configuring MACsec Keychain Policy Using the GUI, on page 235.

Step 2 To apply the **MACsec Access Interface Policy** to a Fabric Leaf or Spine Port Policy Group, in the Navigation pane, click **Interfaces** > ***Leaf/Spine*** **Interfaces** > **Policy Groups** > ***Spine/Leaf*** **Policy Group_*name***. In the **Work** pane, select the **MACsec Fabric Interface Policy** just created.

Configuring MACsec Parameters Using the APIC GUI

Procedure

Step 1 On the menu bar, click **Fabric** > **Access Policies**. In the **Navigation** pane, click on **Interface Policies** > **Policies** and right click on **MACsec Policies** to open **Create MACsec Access Parameters Policy** and perform the following actions:

a) In the **Name** field, enter a name for the MACsec Access Parameters policy.

b) In the **Security Policy** field, select a mode for encrypted traffic and click **Submit**.

Note
Before deploying MACsec in **Must Secure Mode**, the keychain must be deployed on the affected interface or the interface will go down.

Step 2 To apply the **MACsec Access Parameters Policy** to a Leaf or Spine Port Policy Group, in the Navigation pane, click **Interface Policies** > **Policy Groups** > **Spine/Leaf Policy Group_*name***. In the **Work** pane, select the **MACsec Access Interface Policy** just created.

Configuring MACsec Keychain Policy Using the GUI

Procedure

Step 1 On the menu bar, click **Fabric** > **Fabric Policies** > **Policies** > **MACsec** > **KeyChains**. In the **Navigation** pane, right click on **KeyChains** to open **Create MACsec Keychain Policy** and perform the following actions:

a) In the **Name** field, enter a name for the MACsec Fabric Interface policy.

b) Expand the **MACsec Key Policy** table to create the Key policy.

Step 2 In the **MACsec Key Policy** dialog box perform the following actions:

a) In the **Name** field, enter a name for the MACsec Key policy.

b) In the **Key Name** field, enter a key name (up to 64 hexadecimal characters).

Note
A maximum of 64 keys are supported per keychain.

c) In the **Pre-shared Key** field, enter the pre-shared key information.

Note
- For 128-bit cipher suites only 32 character PSKs are permitted.

- For 256-bit cipher suites only 64 Character PSKs are permitted.

d) In the **Start Time** field, select a date for the key to become valid.
e) In the **End Time** field, select a date for the key to expire. Click **Ok** and **Submit**.

Note
When defining multiple keys in a keychain, the keys must be defined with overlapping times in order to assure a smooth transition from the old key to the new key. The endTime of the old key should overlap with the startTime of the new key.

For configuring the Keychain policy through Access Policies, on the menu bar click **Fabric** > **External Access Policies**. In the **Navigation** pane, click on **Policies** > **Interface** > **MACsec** > **MACsec KeyChain Policies** and right click on to open **Create MACsec Keychain Policy** and perform the steps above.

Configuring MACsec Using the NX-OS Style CLI

Procedure

Step 1 Configure MACsec Security Policy for access interfaces

Example:

```
apic1# configure
apic1(config)#   template macsec access security-policy accmacsecpol1
apic1(config-macsec-param)#     cipher-suite gcm-aes-128
apic1(config-macsec-param)#     conf-offset offset-30
apic1(config-macsec-param)#     description 'description for mac sec parameters'
apic1(config-macsec-param)#     key-server-priority 1
apic1(config-macsec-param)#     sak-expiry-time 110
apic1(config-macsec-param)#     security-mode must-secure
aapic1(config-macsec-param)#     window-size 1
apic1(config-macsec-param)#     exit
apic1(config)#
```

Step 2 Configure MACsec key chain for access interface:

PSK can be configured in 2 ways:

Note
- Inline with the **psk-string** command as illustrated in key 12ab below. The PSK is not secure because it is logged and exposed.
- Entered separately in a new command **Enter PSK string** after the **psk-string** command as illustrated in key ab12. The PSK is secured because it is only echoed locally and is not logged.

Example:

```
apic1# configure
apic1(config)#   template macsec access keychain acckeychainpol1
apic1(config-macsec-keychain)#     description 'macsec key chain kc1'
apic1(config-macsec-keychain)#     key 12ab
```

```
apic1(config-macsec-keychain-key)#       life-time start 2017-09-19T12:03:15 end 2017-12-19T12:03:15
apic1(config-macsec-keychain-key)#       psk-string 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)#       exit
apic1(config-macsec-keychain)#     key ab12
apic1(config-macsec-keychain-key)#       life-time start now end infinite
apic1(config-macsec-keychain-key)#       life-time start now end infinite
apic1(config-macsec-keychain-key)# psk-string
Enter PSK string: 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)# exit
apic1(config-macsec-keychain)# exit
apic1(config)#
```

Step 3 Configure MACsec interface policy for access interface:

Example:

```
apic1# configure
apic1(config)#   template macsec access interface-policy accmacsecifpol1
apic1(config-macsec-if-policy)#     inherit macsec security-policy accmacsecpol1 keychain
acckeychainpol1
apic1(config-macsec-if-policy)#     exit
apic1(config)#
```

Step 4 Associate MACsec interface policy to access interfaces on leaf (or spine):

Example:

```
apic1# configure
apic1(config)#   template macsec access interface-policy accmacsecifpol1
apic1(config-macsec-if-policy)#     inherit macsec security-policy accmacsecpol1 keychain
acckeychainpol1
apic1(config-macsec-if-policy)#     exit
apic1(config)
```

Step 5 Configure MACsec Security Policy for fabric interfaces:

Example:

```
apic1# configure
apic1(config)#   template macsec fabric security-policy fabmacsecpol1
apic1(config-macsec-param)#     cipher-suite gcm-aes-xpn-128
apic1(config-macsec-param)#     description 'description for mac sec parameters'
apic1(config-macsec-param)#     window-size 1
apic1(config-macsec-param)#     sak-expiry-time 100
apic1(config-macsec-param)#     security-mode must-secure
apic1(config-macsec-param)#     exit
apic1(config)#
```

Step 6 Configure MACsec key chain for fabric interface:

PSK can be configured in 2 ways:

Note

- Inline with the **psk-string** command as illustrated in key 12ab below. The PSK is not secure because it is logged and exposed.
- Entered separately in a new command **Enter PSK string** after the **psk-string** command as illustrated in key ab12. The PSK is secured because it is only echoed locally and is not logged.

Example:

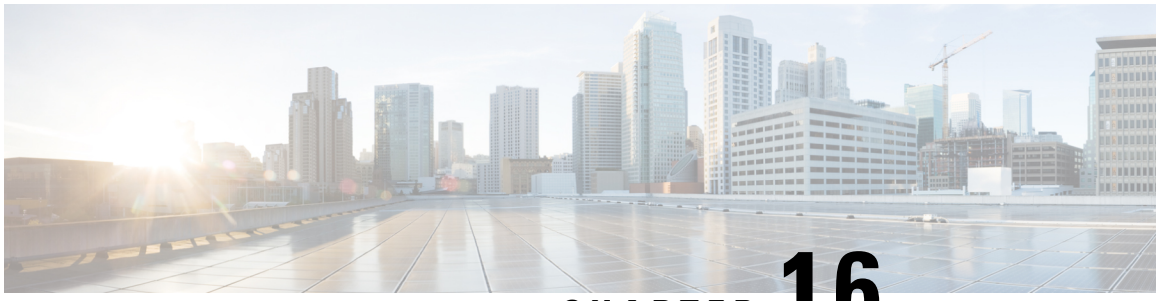
```
apic1# configure
apic1(config)#   template macsec fabric security-policy fabmacsecpol1
apic1(config-macsec-param)#     cipher-suite gcm-aes-xpn-128
apic1(config-macsec-param)#     description 'description for mac sec parameters'
```

```
apic1(config-macsec-param)#     window-size 1
apic1(config-macsec-param)#     sak-expiry-time 100
apic1(config-macsec-param)#     security-mode must-secure
apic1(config-macsec-param)#     exit
apic1(config)#   template macsec fabric keychain fabkeychainpol1
apic1(config-macsec-keychain)#     description 'macsec key chain kc1'
apic1(config-macsec-keychain)#     key 12ab
apic1(config-macsec-keychain-key)#       psk-string 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)#       life-time start 2016-09-19T12:03:15 end 2017-09-19T12:03:15
apic1(config-macsec-keychain-key)#       exit
apic1(config-macsec-keychain)#     key cd78
apic1(config-macsec-keychain-key)# psk-string
Enter PSK string: 123456789a223456789a323456789abc
apic1(config-macsec-keychain-key)# life-time start now end infinite
apic1(config-macsec-keychain-key)# exit
apic1(config-macsec-keychain)# exit
apic1(config)#
```

Step 7 Associate MACsec interface policy to fabric interfaces on leaf (or spine):

Example:

```
apic1# configure
apic1(config)#   leaf 101
apic1(config-leaf)#     fabric-interface ethernet 1/52-53
apic1(config-leaf-if)#       inherit macsec interface-policy fabmacsecifpol2
apic1(config-leaf-if)#       exit
apic1(config-leaf)#
```



CHAPTER 16

Fabric Port Tracking

- About Fabric Port Tracking, on page 239
- Configuring Fabric Port Tracking Using the GUI, on page 240

About Fabric Port Tracking

The port tracking feature manages the status of downlink ports on each leaf node based on the status of its fabric ports. Fabric ports are the links between leaf and spine nodes. Links between tier-1 and tier-2 leaf nodes in multi-tier topologies and links between remote leaf nodes (back-to-back links) are also considered to be fabric links.

When this feature is enabled and the number of operational fabric ports on a given leaf node is decreased to the configured threshold or lower, the downlink ports of the leaf node will be brought down so that external devices can switch over to other healthy leaf nodes. When the number of operational fabric ports comes back up to greater than the configured threshold, the downlink ports will be brought back up. At this time, there will be a wait time of the configured delay before the downlink ports are brought up. If the leaf node is part of a vPC peer and does not have any infra ISIS adjacencies--meaning that the node is unable to communicate with the other vPC peer leaf node--when port tracking is triggered such as when all fabric ports went down, the wait time for the vPC downlink ports to come up after the status restoration will be the longer time of either the vPC delay timer or the configured delay in port tracking. Non-vPC downlink ports always follow the delay timer configured in port tracking.

Beginning in Cisco Application Centric Infrastructure (ACI) switch release 14.2(1), the status of fabric infra ISIS adjacency (represented by the attribute `aggFabAdjOperSt` of the managed object class `isisDom`) is also checked as an alternative condition to trigger the shutdown of downlink ports. The check is performed to cover a scenario in which fabric ports on a given leaf node are up, but the leaf node has lost reachability to other Cisco ACI nodes for another reason. This condition is always checked when the feature is enabled regardless of the other parameters, such as the minimum number of operational fabric ports. However, it is not applicable to remote leaf nodes because those nodes do not rely on ISIS for fabric infra reachability.

Beginning in Cisco ACI switch release 15.0(1), the "Include APIC ports" option is supported. This option is disabled by default. When this option is disabled, port tracking only brings down downlink ports that are configured for user traffic (that is, ports that are in use by EPGs or L3Outs) while downlink ports connected to the Cisco Application Policy Infrastructure Controllers (APICs) or unused ports are not brought down. When this option is enabled, port tracking brings down all downlink ports on the leaf node. Prior to release 15.0(1), downlink ports that were configured for user traffic were brought down while ports connected to the Cisco APIC were not, and you could not configure port tracking to bring down ports that were connected to the Cisco APIC.

Note Port tracking checks the conditions to bring down the ports or bring up the ports every second on each leaf node.

FEX fabric ports--that is, the network interface (NIF) to connect the FEX and the FEX's parent leaf node--are not impacted by port tracking.

Configuring Fabric Port Tracking Using the GUI

This procedure uses the Cisco Application Policy Infrastructure Controller (APIC) GUI to configure the port track feature.

Procedure

Step 1 In the 3.2(1) release or earlier, navigate to the following location:

a) On the menu bar, choose **Fabric** > **Access Policies**.
b) In the Navigation pane, choose **Policies** > **Global**.
c) Click **Port Tracking**.

Step 2 In the 3.2(2) release, navigate to the following location:

a) On the menu bar, choose **System** > **System Settings**.
b) In the Navigation pane, choose **Port Tracking**.

Step 3 For the **Port tracking state** parameter, choose **on** to enable fabric port tracking.

Step 4 For the **Delay restore timer** parameter, specify the time in seconds.

This parameter determines how long the leaf node will wait before bringing up its downlink ports after the fabric port status and infra ISIS adjacencies are restored.

Step 5 Configure the **Number of active fabric ports that triggers port tracking** parameter.

When the number of operational fabric ports on a leaf node is decreased to the configured number or lower, the leaf node will bring down its downlink ports.