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

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

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

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

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

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

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

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

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

© 2016–2020 Cisco Systems, Inc. All rights reserved.
CONTENTS

Preface xv

Audience xv

Document Conventions xv

Related Documentation xvii

Documentation Feedback xviii

Obtaining Documentation and Submitting a Service Request xviii

Overview 1

About This Document 1

Design 3

ACI Constructs Design 5

Common Tenant and User-Configured Tenant Policy Usage 5

About Common Tenant and User-Configured Tenant Policy Usage 5

Prerequisites for Common Tenant and User-Configured Tenant Policy Usage 6

Guidelines and Limitations for Common Tenant and User-Configured Tenant Policy Usage 6

Recommended Configuration Procedure for Common Tenant and User-Configured Tenant Policy Usage 6

Verifying the Common Tenant and User-Configured Tenant Policy Usage 7

Configuration Examples for Common Tenant and User-Configured Tenant Policy Usage 7

Additional References for Common Tenant and User-Configured Tenant Policy Usage 8

Common Pervasive Gateway 8

About Common Pervasive Gateway 8

Prerequisites for Common Pervasive Gateway 9

Guidelines and Limitations for Common Pervasive Gateway 9
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recommended Configuration Procedure for Common Pervasive Gateway</td>
<td>9</td>
</tr>
<tr>
<td>Verifying the Common Pervasive Gateway Using the GUI</td>
<td>10</td>
</tr>
<tr>
<td>Additional References for Common Pervasive Gateway</td>
<td>10</td>
</tr>
<tr>
<td>Contracts and Policy Enforcement</td>
<td>11</td>
</tr>
<tr>
<td>About Contracts and Policy Enforcement</td>
<td>11</td>
</tr>
<tr>
<td>Guidelines and Limitations for Contracts and Policy Enforcement</td>
<td>16</td>
</tr>
<tr>
<td>Additional References for Contracts and Policy Enforcement</td>
<td>17</td>
</tr>
<tr>
<td>Contract Labels</td>
<td>17</td>
</tr>
<tr>
<td>About Contract Labels</td>
<td>17</td>
</tr>
<tr>
<td>Prerequisites for Contract Labels</td>
<td>17</td>
</tr>
<tr>
<td>Guidelines and Limitations for Contract Labels</td>
<td>17</td>
</tr>
<tr>
<td>Recommended Configuration Procedure for Contract Labels</td>
<td>18</td>
</tr>
<tr>
<td>Verifying the Contract Labels Using the GUI</td>
<td>18</td>
</tr>
<tr>
<td>Configuration Examples for Contract Labels</td>
<td>18</td>
</tr>
<tr>
<td>Additional References for Contract Labels</td>
<td>19</td>
</tr>
<tr>
<td>Taboo Contracts</td>
<td>19</td>
</tr>
<tr>
<td>About Taboo Contracts</td>
<td>19</td>
</tr>
<tr>
<td>Prerequisites for Taboo Contracts</td>
<td>20</td>
</tr>
<tr>
<td>Guidelines and Limitations for Taboo Contracts</td>
<td>20</td>
</tr>
<tr>
<td>Recommended Configuration Procedure for Taboo Contracts</td>
<td>20</td>
</tr>
<tr>
<td>Configuration Examples for Taboo Contracts</td>
<td>21</td>
</tr>
<tr>
<td>Additional References for Taboo Contracts</td>
<td>21</td>
</tr>
<tr>
<td>Bridge Domains</td>
<td>21</td>
</tr>
<tr>
<td>About Bridge Domains</td>
<td>21</td>
</tr>
<tr>
<td>Guidelines and Limitations for Bridge Domains</td>
<td>22</td>
</tr>
<tr>
<td>Recommended Configuration Procedure for Bridge Domains</td>
<td>22</td>
</tr>
<tr>
<td>Application-Centric and Network-Centric Deployments</td>
<td>28</td>
</tr>
<tr>
<td>About Application-Centric and Network-Centric Deployments</td>
<td>28</td>
</tr>
<tr>
<td>Layer 2 Extension</td>
<td>31</td>
</tr>
<tr>
<td>About Layer 2 Extension</td>
<td>31</td>
</tr>
<tr>
<td>Configuration Examples for Layer 2 Extension</td>
<td>31</td>
</tr>
<tr>
<td>Additional References for Layer 2 Extension</td>
<td>32</td>
</tr>
<tr>
<td>Infrastructure VXLAN Tunnel Endpoint Pool</td>
<td>33</td>
</tr>
<tr>
<td>About Infrastructure VXLAN Tunnel Endpoint Pool</td>
<td>33</td>
</tr>
</tbody>
</table>
### Contents

- **Layer 4 to Layer 7 Design** 109
  - About Service Graphs and Layer 4 to Layer 7 Services Integration 109
  - Layer 4 to Layer 7 Services Integration Options 109
  - When to Use a Service Graph for Layer 4 to Layer 7 Services Integration 111
  - Service Node Failover 117
  - Service Graphs with Multiple Consumers and Providers 119

- **Cisco Application Centric Infrastructure Best Practices Guide, Release 1.3(1) and Earlier**
  - Chapter 6: Service Graphs and Layer 4 to Layer 7 Services Integration 109
The Basic GUI and the Advanced GUI 183
  About the Basic GUI and the Advanced GUI 183
  Prerequisites for the Basic GUI vs the Advanced GUI 183
  Guidelines and Limitations for Basic GUI vs Advanced GUI 183
  Verifying the Basic GUI vs the Advance GUI 184
  Additional References for Using the Basic GUI and Advanced GUI 184

Migrating Existing Networks to Cisco ACI 184
  About Migrating Existing Networks to Cisco ACI 184
  Prerequisites for Migrating Existing Networks to Cisco ACI 184
  Recommended Configuration Procedure for Migrating Existing Networks to Cisco ACI 185
  Additional References for Migrating Existing Networks to Cisco ACI 185

PART III OPERATIONS 187

CHAPTER 12 ACI Constructs Operations 189
  AAA RBAC and Roles 189
    About AAA RBAC and Roles 189
    Prerequisites for AAA RBAC and Roles 190
    Guidelines and Limitations for AAA RBAC and Roles 190
    Recommended Configuration Procedure for AAA RBAC and Roles 191
    Verifying the AAA RBAC and Roles Using the GUI 191
    Configuration Examples for AAA RBAC and Roles Using the GUI 191
    Additional References for AAA RBAC and Roles 192

  Endpoint Loop Protection 192
    About Endpoint Loop Protection 192
    Configuration Example for Endpoint Loop Protection 193

CHAPTER 13 Layer 4 to Layer 7 Operations 195
  Device Packages 195
    About Device Packages 195
    Guidelines and Limitations for Device Packages 195
    Recommended Procedure for Importing a Device Package Using the GUI 196
    Verifying the Device Package Versions 197
CHAPTER 14 Miscellaneous Operations

API Inspector

About API Inspector
Recommended Configuration Procedure for API Inspector
Verifying an API Inspector Configuration
Configuration Example for API Inspector

Audit Logs

About Audit Logs
Prerequisites for Audit Logs
Guidelines and Limitations for Audit Logs
Verifying the Audit Logs Using the GUI
Verifying Audit Logs Using the Object Model CLI
Additional References for Audit Logs

GUI Application Settings

About the GUI Application Settings
Prerequisites for GUI Application Settings
Guidelines and Limitations for GUI Application Settings
Recommended Configuration Procedure for GUI Application Settings
Configuring GUI Application Settings
Verifying the GUI Application Settings

Health Scores

About Health Score
Prerequisites for Health Score
Guidelines and Limitations for Health Score
Recommended Configuration Procedure for Health Scores
Verifying Health Score
Additional References for Health Score

Using the Cisco NX-OS Style CLI

About Cisco NX-OS Style CLI
Prerequisites for Cisco NX-OS Style CLI
Guidelines and Limitations for Cisco NX-OS Style CLI
Verifying the Cisco NX-OS Style CLI
Configuration Examples for the Cisco NX-OS-Style CLI
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional References for the Cisco NX-OS-Style CLI</td>
<td>208</td>
</tr>
<tr>
<td>Upgrading the Fabric</td>
<td>208</td>
</tr>
<tr>
<td>Guidelines and Limitations for Adding a Switch</td>
<td>208</td>
</tr>
<tr>
<td>Guidelines and Limitations for Upgrading the Fabric</td>
<td>208</td>
</tr>
<tr>
<td>Additional References for Upgrading the Fabric</td>
<td>209</td>
</tr>
<tr>
<td>Snapshot and Configuration Rollback</td>
<td>210</td>
</tr>
<tr>
<td>About Snapshot and Configuration Rollback</td>
<td>210</td>
</tr>
<tr>
<td>Guidelines and Limitations when Using Snapshot and Rollback</td>
<td>210</td>
</tr>
<tr>
<td>Recommended Procedure for Snapshot and Rollback</td>
<td>210</td>
</tr>
<tr>
<td>Configuration Example for Snapshot and Configuration Rollback</td>
<td>210</td>
</tr>
<tr>
<td>Verifying a Snapshot and Rollback Configuration</td>
<td>211</td>
</tr>
<tr>
<td>Additional References for Snapshot and Configuration Rollback</td>
<td>211</td>
</tr>
<tr>
<td>Tags and Aliases</td>
<td>211</td>
</tr>
<tr>
<td>About Tags and Aliases</td>
<td>211</td>
</tr>
<tr>
<td>Guidelines and Limitations for Tags and Aliases</td>
<td>212</td>
</tr>
<tr>
<td>Recommended Configuration Procedures for Tags and Aliases</td>
<td>212</td>
</tr>
<tr>
<td>Verifying Tags and Aliases</td>
<td>212</td>
</tr>
<tr>
<td>Additional References for Tags and Aliases</td>
<td>213</td>
</tr>
<tr>
<td>QuickStart in the Cisco APIC GUI</td>
<td>213</td>
</tr>
<tr>
<td>About QuickStart in the APIC GUI</td>
<td>213</td>
</tr>
<tr>
<td>Prerequisites for QuickStart</td>
<td>213</td>
</tr>
<tr>
<td>Guidelines and Limitations for QuickStart</td>
<td>214</td>
</tr>
<tr>
<td>Configuration Examples for QuickStart</td>
<td>214</td>
</tr>
</tbody>
</table>
Preface

This preface includes the following sections:

- Audience, on page xv
- Document Conventions, on page xv
- Related Documentation, on page xvii
- Documentation Feedback, on page xviii
- Obtaining Documentation and Submitting a Service Request, on page xviii

Audience

This guide is intended primarily for data center administrators with responsibilities and expertise in one or more of the following:

- Virtual machine installation and administration
- Server administration
- Switch and network administration
- Cloud administration

Document Conventions

Command descriptions use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bold</td>
<td>Bold text indicates the commands and keywords that you enter literally as shown.</td>
</tr>
<tr>
<td>Italic</td>
<td>Italic text indicates arguments for which the user supplies the values.</td>
</tr>
<tr>
<td>[x]</td>
<td>Square brackets enclose an optional element (keyword or argument).</td>
</tr>
<tr>
<td>[x</td>
<td>y]</td>
</tr>
</tbody>
</table>
**Conventions**

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>{x</td>
<td>y}</td>
</tr>
<tr>
<td>[x {y</td>
<td>z}]</td>
</tr>
<tr>
<td>variable</td>
<td>Indicates a variable for which you supply values, in context where italics cannot be used.</td>
</tr>
<tr>
<td>string</td>
<td>A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.</td>
</tr>
</tbody>
</table>

Examples use the following conventions:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>screen</strong> <strong>font</strong></td>
<td>Terminal sessions and information the switch displays are in screen font.</td>
</tr>
<tr>
<td><strong>boldface</strong> <strong>screen</strong> <strong>font</strong></td>
<td>Information you must enter is in boldface screen font.</td>
</tr>
<tr>
<td><strong>italic</strong> <strong>screen</strong> <strong>font</strong></td>
<td>Arguments for which you supply values are in italic screen font.</td>
</tr>
<tr>
<td>&lt; &gt;</td>
<td>Nonprinting characters, such as passwords, are in angle brackets.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Default responses to system prompts are in square brackets.</td>
</tr>
<tr>
<td>!, #</td>
<td>An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.</td>
</tr>
</tbody>
</table>

This document uses the following conventions:

**Note**

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

**Caution**

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

**Warning**

**IMPORTANT SAFETY INSTRUCTIONS**

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

**SAVE THESE INSTRUCTIONS**
Related Documentation

**Cisco Cloud APIC Documentation**


**Cisco Application Policy Infrastructure Controller (APIC) Documentation**

The following companion guides provide documentation for Cisco APIC:

- *Cisco APIC Getting Started Guide*
- *Cisco APIC Basic Configuration Guide*
- *Cisco ACI Fundamentals*
- *Cisco APIC Layer 2 Networking Configuration Guide*
- *Cisco APIC Layer 3 Networking Configuration Guide*
- *Cisco APIC NX-OS Style Command-Line Interface Configuration Guide*
- *Cisco APIC REST API Configuration Guide*
- *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide*
- *Cisco ACI Virtualization Guide*
- *Cisco Application Centric Infrastructure Best Practices Guide*

All these documents are available at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html

**Cisco Application Centric Infrastructure (ACI) Documentation**


**Cisco Application Centric Infrastructure (ACI) Simulator Documentation**


**Cisco Nexus 9000 Series Switches Documentation**

Cisco ACI Virtual Edge Documentation


Cisco ACI Virtual Pod Documentation


Cisco Application Centric Infrastructure (ACI) Integration with OpenStack Documentation


Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to apic-docfeedback@cisco.com. We appreciate your feedback.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see What’s New in Cisco Product Documentation at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html

Subscribe to What’s New in Cisco Product Documentation, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.
Overview

• About This Document, on page 1

About This Document

This document provides best practices for the design, implementation, and operation of Cisco Application Centric Infrastructure (ACI). The best practices apply to some of the more common use cases of ACI; it is impossible to provide information about every possible configuration of ACI.

This document applies to the ACI 1.3(1) release and prior.
PART I

Design

• ACI Constructs Design, on page 5
• Routing Design, on page 49
• Security Design, on page 83
• Virtualization Design, on page 91
• Layer 4 to Layer 7 Design, on page 109
• Miscellaneous Design, on page 139
Common Tenant and User-Configured Tenant Policy Usage

About Common Tenant and User-Configured Tenant Policy Usage

A tenant is a logical container for application, networking and security policies. The rules governing policy reuse across tenants differ between user-configured tenants and the system-defined common tenant.

An example would be that user-configured tenant "A" has a bridge domain, while user-configured tenant "B" has an endpoint group. By default, tenant B's endpoint group will never be able to make an association to tenant A's bridge domain. Objects within user-configured tenants cannot form relationships with objects in other user-configured tenants unless specified with explicit configurations. One example of this is the process of exporting a contract from one user-configured tenant to another. Otherwise, that contract can only be referenced by other objects within the same tenant.

When utilizing the system-generated tenant common, this rule does not apply. Objects within tenant common can be accessed by all other tenants within a Cisco Application Centric Infrastructure (ACI) fabric. This means that tenant B's endpoint group would be able to use a bridge domain configured within tenant common. Similarly, tenant B's endpoint group would be able to use a contract that exists within tenant common without needing to be exported.
**Prerequisites for Common Tenant and User-Configured Tenant Policy Usage**

You must meet the following prerequisites to use the common tenant and user-configured tenant policies:

- Tenant common is system generated and has no prerequisite configuration to allow its policies to be accessed by other tenants.

- A user-configured tenant must be created before usage. Not all user-configured tenant policies can be made accessible to other tenants. The following policies can be exported from one user-configured tenant to another to form a relationship:
  
  - Contracts
  
  - Layer 4 to Layer 7 devices

**Guidelines and Limitations for Common Tenant and User-Configured Tenant Policy Usage**

The following guidelines and limitations apply for common tenant and user-configured tenant policy usage:

- There are specific policies within a user-configured tenant that can be exported to another tenant for relationship usage.

- A VRF named "myVRF" within user-configured tenant A is not the same as a VRF named "myVRF" within user-configured tenant B. This difference can be observed by looking at the distinguished name (DN) of both VRFs.

- Depending on the intended usage of these exported policies, there might be other configuration changes required to complete inter-tenant communication. For more information, see About Shared Services, on page 153.

**Recommended Configuration Procedure for Common Tenant and User-Configured Tenant Policy Usage**

The following procedure exports contracts and Layer 4 to Layer 7 devices from a user-configured tenant using the Application Policy Infrastructure Controller (APIC) GUI, which you can then import into another user-configured tenant. You must use the advanced GUI mode.

**Procedure**

1. **Step 1** Export a contract. On the menu bar, choose **Tenants > All Tenants**.
2. **Step 2** In the **Work** pane, double-click the desired tenant's name.
3. **Step 3** In the **Navigation** pane, choose **Tenant tenant_name > Security Policies > Contracts**.
4. **Step 4** In the **Work** pane, choose **Actions > Export Contract**.
5. **Step 5** In the **Export Contract** dialog box, fill out the fields as necessary.

For a contract to be used between endpoint groups within separate VRFs, the contract scope must be changed to **Global**. The scope is set to **VRF** by default.
Step 6 Export a Layer 4 to Layer 7 device. On the menu bar, choose Tenants > All Tenants.

Step 7 In the Work pane, double-click the user-configured tenant's name from which you will export the contract.

Step 8 In the Navigation pane, choose Tenant tenant_name > L4-L7 Services > L4-L7 Devices.

Step 9 In the Work pane, choose Actions > Export L4-L7 Devices.

Step 10 In the Export L4-L7 Devices dialog box, fill out the fields as necessary.

Verifying the Common Tenant and User-Configured Tenant Policy Usage

A general guide to understanding where a policy resides is to understand the distinguished name (DN) of that object. This can be said for almost every policy within Cisco Application Centric Infrastructure (ACI), but especially so for those configured within tenants. Most objects in the GUI allow you to right-click on them and choose Save As. This will allow you to pull either an XML or JSON representation of the object you chose, and potentially its children objects as well if desired.

The following procedure provides an example of saving a contract named "BP-contract" that was created in the tenant "ACI-BP":

Procedure

Step 1 On the menu bar, choose Tenants > All Tenants.
Step 2 In the Work pane, double-click ACI-BP.
Step 3 In the Navigation pane, choose Tenant ACI-BP > Security Policies > Contracts > BP-contract.
Step 4 Right-click the contract and choose Save as ....
Step 5 In the Save As dialog box, click Only Configuration, Self, and xml.
Step 6 Click Download.

The saved XML file contains the following lines:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="1">
  <vzBrCP scope="context" prio="unspecified" ownerTag="" ownerKey="" name="BP-contract" dn="uni/tn-ACI-BP/brc-BP-contract" descr=""/>
</imdata>
```

The dn parameter has a value of "uni/tn-ACI-BP/brc-BP-contract." Without examining the classes, you can see that this contract exists directly under tenant ACI-BP and that the contract name is "BP-contract."

Configuration Examples for Common Tenant and User-Configured Tenant Policy Usage

When selecting a policy for use, you can typically see the tenant association during the selection process. For example, when attempting to associate a contract to an endpoint group within a user-configured tenant, a variety of contract choices might display, such as in the following example list:

- multiservice/CTRCT1
The contract naming convention is "tenant/contract_name." From the example contract names, you can infer that all choices that begin with "common/" exist within the common tenant, while all choices prefixed with "multiservice/" have been created within the user-configured tenant "multiservice."

Additional References for Common Tenant and User-Configured Tenant Policy Usage

For more information about tenants, see the Cisco Application Centric Infrastructure (ACI) policy model chapter in the Cisco Application Centric Infrastructure Fundamentals Guide.

Common Pervasive Gateway

About Common Pervasive Gateway

Multiple Cisco Application Centric Infrastructure (ACI) fabrics can be configured with an IPv4 common gateway on a per-bridge-domain basis. Doing so enables moving one or more virtual machines (VMs) or conventional hosts across the fabrics while the host retains its IP address. VM host moves across fabrics can be done automatically by the VM hypervisor. The ACI fabrics can be co-located, or provisioned across multiple sites. The Layer 2 connection between the ACI fabrics can be a local link, or can be across a routed WAN link. The following figure illustrates the basic common pervasive gateway topology:

*Figure 1: Common Pervasive Gateway Topology*
Prerequisites for Common Pervasive Gateway

You must meet the following prerequisites to use common pervasive gateway (CPG):

- Subnets should be determined for CPG
- Common vMAC and unique pMACs across fabrics should be determined
- Hosts to utilize CPG should be set to use the VIP gateway address
- Layer 2 connectivity between fabrics should be established

Guidelines and Limitations for Common Pervasive Gateway

The following guidelines and limitations apply for common pervasive gateway (CPG):

- The bridge domain MAC (pMAC) values for each fabric must be unique.

  The default bridge domain MAC (pMAC) address values are the same for all Cisco Application Centric Infrastructure (ACI) fabrics. The common pervasive gateway requires an administrator to configure the bridge domain MAC (pMAC) values to be unique for each Cisco ACI fabric.

- The bridge domain virtual MAC (vMAC) address and the subnet virtual IP address must be the same across all Cisco ACI fabrics for that bridge domain. Multiple bridge domains can be configured to communicate across connected Cisco ACI fabrics. The virtual MAC address and the virtual IP address can be shared across bridge domains.

- With switch models prior to the "EX" switches, for endpoints residing off bridge domains with a CPG, the fabric will only route traffic that hits the bridge domain by utilizing the vMAC. Any traffic utilizing the pMAC upon entry of the Cisco ACI fabric that is destined for an EP will not be routed. This is normally not a concern if the source device is utilizing ARP lookups before sending a reply, as the gateway entry for the end device should be the VIP/vMAC combo. Traffic sourced from the Cisco ACI bridge domain will always exit the fabric by utilizing the pMAC, not the vMAC. This will cause certain appliances to have communication issues when utilizing specific forwarding features that bypass ARP lookup and instead use the src_mac as the dst_mac in the reply. The following list contains examples of features that bypass ARP lookup:
  - EMC "Packet Reflect"
  - F5 "Auto Last Hop"
  - Netapp "Fast Path"

Recommended Configuration Procedure for Common Pervasive Gateway

The following information applies when configuring common pervasive gateway (CPG):

- Ensure that all end devices utilizing a CPG as its gateway should perform ARP lookups in all communication scenarios. Any device that utilizes some feature that bypasses this lookup will have communication issues when trying to get to another subnet within the fabric.

- The pMAC for bridge domains across two separate Cisco Application Centric Infrastructure (ACI) fabrics are unique.
• The vMAC across matching bridge domains should be configured the same across both ACI fabrics that utilize CPG.

• The VIP address will be set as a virtual IP and will act as the gateway for hosts within this subnet.

**Verifying the Common Pervasive Gateway Using the GUI**

The following procedure verifies the common pervasive gateway (CPG) configuration using the Application Policy Infrastructure Controller (APIC) GUI.

**Procedure**

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>On the menu bar, choose <strong>Tenants &gt; All Tenants</strong>.</td>
</tr>
<tr>
<td>Step 2</td>
<td>In the <strong>Work</strong> pane, double-click the desired tenant's name.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the <strong>Navigation</strong> pane, choose <strong>Tenant tenant_name &gt; Networking &gt; Bridge Domains &gt; bridge_domain_name</strong>.</td>
</tr>
<tr>
<td>Step 4</td>
<td>In the <strong>Work</strong> pane, choose the <strong>Policy &gt; L3 Configurations</strong> tabs.</td>
</tr>
</tbody>
</table>

The Work pane displays the configuration pieces that are needed for a common pervasive gateway.

**Step 5**

The **Custom MAC Address** field is the pMAC that must be unique between both Cisco Application Centric Infrastructure (ACI) fabrics sharing the CPG. By default, all ACI fabrics have the same value. If the value is the same for both fabrics, change the value either of the fabrics.

**Step 6**

The **Virtual MAC Address** field is the vMAC that must be the same between both bridge domains across both ACI fabrics. Replace the “Not Configured” text with a valid MAC address.

**Step 7**

Put a check in the **Treat as virtual IP address** check box to define the subnet to be the VIP address under the bridge domain.

This should be done for the address that will be shared across both bridge domains and act as the GW for hosts on this subnet. Otherwise, another subnet/bridge domain address will need to be created that is unique to this fabric. For example, assume that 192.168.1.1 will be the VIP and exist as the virtual IP address on both fabrics' bridge domains. Fabric 1 will have a second subnet under the bridge domain set as 192.168.1.2, and Fabric 2 will have a second subnet under the bridge domain set as 192.168.1.3. These second subnets will not be virtual IPs, but instead will act as the bridge domain SVI.

**Additional References for Common Pervasive Gateway**

For more information on the common pervasive gateway traffic flow, see the tenants chapter of the *Operating Cisco Application Centric Infrastructure* document at the following URL:

Contracts and Policy Enforcement

About Contracts and Policy Enforcement

Contracts

By default, a VRF is in enforced mode, which means that without a contract, different endpoint groups are unable to communicate to each other. Endpoint groups associate to a contract with provider/consumer relationships. ACLs, rules, and filters are created in the leaf switches to realize the intent of contracts that will be programmed on the ternary content-addressable memory (TCAM). The following figure illustrates endpoint groups communicating through contracts:

Figure 2: Endpoint Group Communication Through Contracts

Policy information in Cisco Application Centric Infrastructure (ACI) is programmed into two TCAM tables:

- Policy TCAM contains entries for the allowed endpoint-group-to-endpoint-group traffic
- App TCAM contains shared destination Layer 4 port ranges

The size of the policy TCAM depends on the generation of Cisco ASIC that is in use. For ALE-based systems, the policy TCAM size is 4k entries. For ALE2-based systems, 32k hardware entries are available. In certain larger scale environments, it is important to take policy TCAM usage into account and ensure that the limits are not exceeded.

TCAM entries are generally specific to each endpoint group pair. In other words, even if the same contract is reused, new TCAM entries are installed for every pair of endpoint groups, as shown in the following figure:
An approximate calculation for the number of TCAM entries is as follows:

\[
\text{Number of entries in a contract} \times \text{Number of Consumer EPGs} \times \text{Number of Provider EPGs} \times 2
\]

**vzAny**

The "Any" endpoint group is a collection of all of the endpoint groups within a context, which is also known as a virtual routing and forwarding (VRF), that allows for a shorthand way to refer to all of the endpoint groups within that context. This shorthand referral eases management by allowing for a single point of contract configuration for all endpoint groups within a context, and also optimizes hardware resource consumption by applying the contract to this one group rather than to each endpoint group individually.

Consider the example shown in the following figure:

**Figure 4: Multiple Endpoint Groups Consuming a Single Contract**

In this scenario, a single endpoint group named "Shared" is providing a contract, with multiple endpoint groups consuming that contract. Although this setup works, it has some drawbacks. First, the administrative burden increases, as each endpoint group must be configured separately to consume the contract. Second, the number of hardware TCAM entries increases each time an endpoint group associates with a contract. A very high number of endpoint groups all providing or consuming a contract can, in extreme cases, lead to exhaustion of the hardware resources.
To overcome these issues, the "vzAny" object can be used. vzAny is a managed object within Cisco Application Centric Infrastructure (ACI) that represents all endpoint groups within a VRF. This object can be used to provide or consume contracts, so in the example above, you can consume the contract from vzAny with the same results, as shown in the following figure:

![vzAny Consuming a Contract](image)

This is not only easier to configure (although automation can eliminate this benefit), but also represents the most efficient use of fabric hardware resources, so is recommended to be used in cases where every endpoint group within a VRF must consume or provide a given contract.

Whenever the use of the vzAny object is being considered, the administrator must plan for its use carefully. Once the vzAny object is configured to provide or consume a contract, any new endpoint groups that are associated with the VRF will inherit the policy; a new endpoint group added to the VRF will provide or consume the same contracts that are configured under vzAny. If it is likely that new endpoint groups will need to be added later and which might not need to consume the same contract as every other endpoint group in the VRF, then vzAny might not be the most suitable choice. You should carefully consider this situation before you use vzAny.

To apply a contract to the vzAny group, choose a tenant in the Application Policy Infrastructure Controller (APIC) GUI. In the Navigation pane, navigate to Tenant > Networking > VRFs > vrf_name > EPG Collection for Context. vrf_name is the name of the VRF for which you want to configure vzAny. EPG Collection for Context is the vzAny object; contracts can be applied here.

**Using vzAny with the "Established Flag"**

An additional example of the use of the vzAny policy to reduce resource consumption is to use it in conjunction with the "established" flag. By doing so, you can configure contracts as unidirectional in nature, which further reduces hardware resource consumption.

Consider the example shown in the following figure:
In this example, two contracts are configured for SSH and HTTP. Both contracts are provided by EPG2 and consumed by EPG1. The **Apply Both Directions** and **Reverse Filter Ports** options are checked, resulting in the four TCAM entries shown in the figure.

You can reduce the TCAM utilization by half by making the contract unidirectional, as shown in the following figure:
However, having a unidirectional contract presents a problem: return traffic is not allowed in the contract, and therefore the connections cannot be completed and traffic fails. To allow return traffic to pass, you can configure a rule that allows traffic between all ports with the "established" flag. We can take advantage of vzAny in this case to configure a single contract for the "established" traffic and apply it to the entire VRF, as shown in the following figure:
In an environment with a large number of contracts being consumed and provided, this can reduce the number of TCAM entries significantly.

**Ingress Policy Enforcement for Border Leaf TCAM Scalability**

Software release 1.2 introduced a new policy enforcement model whereby security rules for all flows are enforced on the leaf node to which internal hosts are connected, rather than at the border leaf. This results in a more even distribution of security rules, rather than being concentrated at the border leaf as was the case prior to release 1.2.

For more information, see About L3Out Ingress Policy Enforcement, on page 64.

**Guidelines and Limitations for Contracts and Policy Enforcement**

The following guidelines and limitations apply when using a vzAny contract:

- When vzAny is used with a contract with scope = Application-Profile, this setting causes rule expansion in the leaf switches and therefore is not recommended.
- vzAny is supported as a consumer of a shared service, but is not supported as a provider of a shared service.
- vzAny is used only to optimize the specification of a source endpoint group or destination endpoint group, by specifying a wildcard for either or both endpoint groups.
If there are ranges in the filter with a \texttt{vzAny} contract, the port range will be done in TCAM to implement the ranges.

**Additional References for Contracts and Policy Enforcement**

For more information about contracts, including procedures for administering contracts, see the *Operating Cisco Application Centric Infrastructure* document at the following URL:


**Contract Labels**

**About Contract Labels**

Contracts are key objects within the Cisco Application Centric Infrastructure (ACI) policy model to express intended communication flows. Endpoint groups can only communicate with other endpoint groups according to the contract rules. A contract can be thought of as an ACL that opens ports between endpoint groups. An administrator uses a contract to select the types of traffic that can pass between endpoint groups, including the protocols and ports allowed. If there are no contracts connecting two endpoint groups, inter-endpoint group communication is disabled by default as long as the VRF is set to \texttt{Enforced}. This is a representation of the white-list policy model that ACI is built around. There is no contract required for intra-endpoint group communication; intra-endpoint group communication is always implicitly allowed regardless of VRF settings. There are configurations that can block intra-endpoint group communication, but is provided by microsegmentation and is not covered in this section.

Contracts can contain multiple communication rules, and multiple endpoint groups can both consume and provide multiple contracts. Labels allow for control over which subjects and filters to apply when communicating between a specific pair of endpoint groups. Without labels, a contract will apply every subject and filter between consumer and provider endpoint groups. A policy designer can use labels to compactly represent a complex communication scenario, within the scope of a single contract, then re-use this contract while specifying only a subset of its policies across multiple endpoint groups.

**Prerequisites for Contract Labels**

You must meet the following prerequisites to use contract labels:

- Contracts should be configured
- Depending on the type of matching to be done, the contract can contain multiple subjects (for subject labels to be useful)
- Have an understanding of the scope of the contract and how to change that setting (the default is \texttt{VRF})

**Guidelines and Limitations for Contract Labels**

The following guidelines and limitations apply for contract labels:
• Understand the scope of a label. Labels can be applied to a variety of provider and consumer managed objects. This includes endpoint groups, contracts, bridge domains, DHCP relay policies, and DNS policies. Labels do not apply across object types; a label on an application endpoint group has no relevance to a label on a bridge domain.

• Labels are managed objects with only one property: a name. Labels enable the classification of which objects can and cannot communicate with one another. Label matching is done first. If the labels do not match, no other contract or filter information is processed.

• Label matching can be applied based on logical operators. The label match attribute can be one of these values: at least one (the default), all, none, or exactly one.

• Because labels are named references, do not to use duplicate label names unless the intent is to chain those flows together.

Recommended Configuration Procedure for Contract Labels

In general, contract labels are not required for contract deployments. For these general scenarios, a single flow can be presented per contract (single subject/group of filters specific to that flow). Utilizing labels does not save resources compared to defining distinct contracts; labels are only another method available to provision contracts while defining specific flows.

Verifying the Contract Labels Using the GUI

The following procedure verifies the programmed rules of a contract under the EPG by using the Application Policy Infrastructure Controller (APIC) GUI. You can use either the advanced basic GUI mode.

Procedure

Step 1 On the menu bar, choose Tenants > All Tenants.
Step 2 In the Work pane, double-click the tenant's name.
Step 3 In the Navigation pane, choose Tenant tenant_name > Application Profiles > application_profile_name > Application EPGs > EPG EPG_name.
Step 4 In the Work pane, choose the Operational > Contracts tabs.

The Work pane displays programmed rules for the contracts. You can ensure that the contract labels are configured properly.

Configuration Examples for Contract Labels

The following procedure provides an example of configuring contract labels using the Application Policy Infrastructure Controller (APIC) GUI.
### Procedure

**Step 1** Configure contract labels (consumer and provider). On the menu bar, choose Tenants > All Tenants.

**Step 2** In the Work pane, double-click the tenant's name.

**Step 3** In the Navigation pane, choose Tenant tenant_name > Security Policies > Contracts > contract_name > contract_subject_name.

**Step 4** In the Work pane, choose the Policy > Label tabs.

The Work pane displays the existing consumed and provided contract labels, and you can configure new labels.

**Step 5** Configure endpoint group subject labels. In the Navigation pane, choose Tenant tenant_name > Application Profiles > application_profiles_name > Application EPGs > EPG EPG_name.

**Step 6** In the Work pane, choose the Policy > Subject Labels tabs.

The Work pane displays the existing consumed and provided endpoint group subject labels, and you can configure new labels.

**Step 7** Configure an endpoint group label when associating a contract as a consumer or provider. In the Navigation pane, choose Tenant tenant_name > Application Profiles > application_profiles_name > Application EPGs > EPG EPG_name > Contracts.

**Step 8** In the Work pane, choose Action > Add Provided Contract or Action > Add Consumed Contract.

**Step 9** In the Add Provided Contract or Add Consumed Contract dialog box, fill out the fields as appropriate and specify the contract label and subject label.

### Additional References for Contract Labels

For more information about contracts and contract labels, see the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:


For more information about Application Policy Infrastructure Controller (APIC) policy enforcement, see the Cisco Application Policy Infrastructure Controller Data Center Policy Model white paper at the following URL:


### Taboo Contracts

#### About Taboo Contracts

Taboo contracts are special contract managed objects in the model that the network administrator can use to deny specific classes of traffic. Taboos can be used to drop traffic matching a pattern, such as any endpoint
group, a specific endpoint group, or matching results from a filter. Taboo rules are applied in the hardware before the rules of regular contracts are applied.

**Prerequisites for Taboo Contracts**

Taboo contracts do not have any specific prerequisites that you must meet.

**Guidelines and Limitations for Taboo Contracts**

In general, the use case for taboo contracts are very specialized and are not seen in a typical deployment. Due to the whitelist nature of Cisco Application Centric Infrastructure (ACI), all flows are blocked by default and those that are to be allowed will need to be specified by a consumer/provider contract relationship.

**Recommended Configuration Procedure for Taboo Contracts**

The following procedure configures a taboo contract.

**Procedure**

<table>
<thead>
<tr>
<th>Step</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Configure a taboo contract within the security policies of a tenant. On the menu bar, choose <strong>Tenants &gt; All Tenants</strong>.</td>
</tr>
<tr>
<td>Step 2</td>
<td>In the <strong>Work</strong> pane, double-click the desired tenant's name.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the <strong>Navigation</strong> pane, choose <strong>Tenant tenant_name &gt; Security Policies &gt; Taboo Contracts</strong>.</td>
</tr>
<tr>
<td>Step 4</td>
<td>In the <strong>Work</strong> pane, choose <strong>Action &gt; Create Taboo Contract</strong>.</td>
</tr>
<tr>
<td>Step 5</td>
<td>In the <strong>Create Taboo Contract</strong> dialog box, fill in the fields as necessary. You must specify the <strong>Name</strong> and add at least one subject. The subject determines what flow to deny explicitly when the taboo contract is applied.</td>
</tr>
<tr>
<td>Step 6</td>
<td>Add a taboo contract to an endpoint group. In the <strong>Navigation</strong> pane, choose <strong>Tenant tenant_name &gt; Application Profiles &gt; application_profile_name &gt; Application EPGs &gt; EPG_name &gt; Contracts</strong>.</td>
</tr>
<tr>
<td>Step 7</td>
<td>In the <strong>Work</strong> pane, choose <strong>Action &gt; Add Taboo Contract</strong>.</td>
</tr>
<tr>
<td>Step 8</td>
<td>In the <strong>Add Taboo Contract</strong> dialog box, choose an existing taboo contract or create a new taboo contract. When adding a taboo contract to an endpoint group, there is no consumer/provider relationship needed to complete the contract flow. The taboo contract will insert a deny specific to that endpoint group once it has been associated to an endpoint group.</td>
</tr>
<tr>
<td>Step 9</td>
<td>(Optional) If you are creating a new taboo contract, in the <strong>Create Taboo Contract</strong> dialog box, fill in the fields as necessary. You must specify the <strong>Name</strong> and add at least one subject. The subject determines what flow to deny explicitly when the taboo contract is applied.</td>
</tr>
</tbody>
</table>
Configuration Examples for Taboo Contracts

One scenario in which taboo contracts can be used is while defining subnets under an L3Out, specifically in the case that subnets are to be blocked. Generally speaking, for an L3Out, the first subnet to be defined is 0.0.0.0/0 as the network, which allows all subnets into the fabric given proper configuration, although this definition is not required. If there are specific subnets for which we want to restrict access into the fabric from this L3Out, you can do so by creating another network under the same L3Out, specifying the subnet to be blocked, then associating the subnet with a taboo contract.

Additional References for Taboo Contracts

For more information on taboo contract fundamentals, see the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:


Bridge Domains

About Bridge Domains

Within a private network, one or more bridge domains must be defined. A bridge domain is a Layer 2 forwarding construct within the fabric, used to constrain broadcast and multicast traffic.

Bridge domains in Cisco Application Centric Infrastructure (ACI) have a number of configuration options to allow the administrator to tune the operation in various ways. The configuration options are as follows:

• L2 Unknown Unicast—This option can be set to either Flood or Hardware Proxy. If this option is set to Flood, Layer 2 unknown unicast traffic will be flooded inside the fabric. If the Hardware Proxy option is set, the fabric mapping database will be queried for Layer 2 unknown unicast traffic. This option does not have any impact on what the mapping database actually learns; the mapping database is always populated for Layer 2 entries regardless of this configuration.

• ARP Flooding—If ARP flooding is enabled, ARP traffic will be flooded inside the fabric as per regular ARP handling in traditional networks. If this option is disabled, the fabric will attempt to unicast the ARP traffic to the destination. This option only applies if unicast routing is enabled on the bridge domain. If unicast routing is disabled, ARP traffic is always flooded, regardless of the status of the ARP Flooding option.

• Unicast Routing—This option enables the learning of IP addresses on the bridge domain in the endpoint table. MAC addresses are always learned by the endpoint table. Using the unicast routing option may be required for some advanced functionality, such as dynamic endpoint attachment with Layer 4 to Layer 7 services. Enabling unicast routing helps to reduce flooding in a bridge domain, as disabling ARP flooding depends upon it. When considering unicast routing, you must consider the desired topology. If an external device (such as a firewall) is acting as the default gateway and there is routing between two bridge domains, enabling unicast routing might cause traffic to be routed on the fabric and bypass the external device. Therefore, as a general best practice, we recommend that you disable unicast routing in a bridge domain that only handles Layer 2 traffic, which is a so-called Layer 2 bridge domain.

• Enforce Subnet Check for IP Learning—If this option is checked, the fabric will not learn IP addresses from a subnet other than the one configured on the bridge domain. For example, if a bridge domain is
configured with a subnet address of 10.1.1.0/24, the fabric would not learn the IP address of an endpoint by using an address that is outside of this range, such as 20.1.1.1/24. This feature does not affect the data path; in other words, it will not drop packets coming from the wrong subnet. The feature simply prevents the fabric from learning endpoint information in this scenario.

Given the above options, it might not be immediately obvious how a bridge domain should be configured. The following sections explain when and why particular options should be selected.

**Guidelines and Limitations for Bridge Domains**

A bridge domain can contain multiple subnets. When you configure a bridge domain with multiple subnets, the first subnet added becomes the primary IP address on the SVI interface. Subsequent subnets are configured as secondary IP addresses. When the switch reloads, the primary IP address might change unless it is marked explicitly.

When using a DHCP relay configuration for bridge domains with multiple subnets, DHCP relay policy can only be configured for the primary IP address on the SVI interface.

If there are DHCP clients that use multiple subnets, make sure you define different bridge domains with each subnet to accommodate that requirement.

To configure a bridge domain subnet as primary, view the subnet's properties and do the following things:

- Put a check in the **Make this IP address primary** check box.

**Recommended Configuration Procedure for Bridge Domains**

The following sections provide the recommended settings for common bridge domain scenarios.

**Scenario 1: IP Address-Based Routed Traffic**

In this scenario, the bridge domain has the following configuration:

- IP address-based routed traffic
- Firewalls and load balancers cannot be connected to this bridge domain
- The bridge domain cannot have clusters or similar things that might rely on "floating" IP addresses (that is, IP addresses that might move to different MACs)
- Silent hosts are not expected to be connected to the bridge domain

Given the above requirements, the recommended bridge domain settings are as follows:

- L2 Unknown Unicast—Hardware Proxy
- Unicast Routing—Enabled
- ARP Flooding—Disabled
- Subnet Configured—Yes, if required
- Enforce Subnet Check for IP Learning—Yes
In this scenario, most of the bridge domain settings can be left at their default, optimized values. A subnet (that is, a gateway address) should be configured as required and you should enforce the subnet check for IP learning.

**Scenario 2: IP Address-Based Routed Traffic, Possible Silent Hosts**

In this scenario, the bridge domain has the following configuration:

- IP address-based routed traffic
- Firewalls and load balancers cannot be connected to this bridge domain
- The bridge domain cannot have clusters or similar things that might rely on "floating" IP addresses (that is, IP addresses that might move to different MACs)
- There might be silent hosts connected to the bridge domain

Given the above requirements, the recommended bridge domain settings are as follows:

- L2 Unknown Unicast—Hardware Proxy
- Unicast Routing—Enabled
- ARP Flooding—Disabled
- Subnet Configured—Yes
- Enforce Subnet Check for IP Learning—Yes

The bridge domain settings for this scenario are similar to scenario 1; however, in this case the subnet address should be configured. As silent hosts can exist within the bridge domain, a mechanism must exist to ensure those hosts are learned correctly inside the Cisco Application Centric Infrastructure (Cisco ACI) fabric. Cisco ACI implements an ARP gleaning mechanism that allows the spine switches to generate an ARP request for an endpoint using the subnet IP address as the source address. This ARP gleaning mechanism ensures that silent hosts are always learned, even when using optimized bridge domain features such as hardware proxy.

The following figure shows the ARP gleaning mechanism when endpoints are not present in the mapping database:
If a subnet IP address cannot be configured for any reason, ARP flooding should be enabled as an alternative to allow the silent hosts to be learned.

**Scenario 3: Non-IP Address-Based Switched Traffic, Possible Silent Hosts**

In this scenario, the bridge domain has the following configuration:

- Non-IP address-based switched traffic
- Firewalls and load balancers cannot be connected to this bridge domain
- The bridge domain cannot have clusters or similar things that might rely on "floating" IP addresses (that is, IP addresses that might move to different MACs)
- There might be silent hosts connected to the bridge domain

Given the above requirements, the recommended bridge domain settings are as follows:

- L2 Unknown Unicast: Flood
- Unicast Routing: Disabled
- ARP Flooding: N/A (enabled automatically due to no unicast routing)
- Subnet Configured: No
- Enforce Subnet Check for IP Learning: N/A

In this scenario, all optimizations inside the bridge domain are disabled and the bridge domain is operating in a "traditional" manner. Silent hosts are dealt with through normal ARP flooding, which is always enabled when unicast routing is turned off.

Also, when operating the bridge domain in a "traditional" mode, the size of the bridge domain should be kept manageable. That is, limit the subnet size and number of hosts as you would in a regular VLAN environment.
**Scenario 4: Non-IP Address or IP Address-Based, Routed or Switched Traffic, Possible "Floating" IP Addresses**

In this scenario, the bridge domain has the following configuration:

- IP address-based or non-IP address-based routed or switched traffic
- Firewalls and load balancers cannot be connected to this bridge domain
- Hosts or devices where the IP address might "float" between MAC addresses
- Silent hosts are not expected to be connected to the bridge domain

Given the above requirements, the recommended bridge domain settings are as follows:

- L2 Unknown Unicast: Hardware Proxy
- Unicast Routing: Enabled
- ARP Flooding: Enabled
- Subnet Configured: Yes
- Enforce Subnet Check for IP Learning: Yes

In this scenario, the bridge domain contains devices where the IP address might move from one device to another, meaning that the IP address moves to a new MAC address. This might be the case where routed firewalls are operating in active/standby mode, or where server clustering is used. Where this is a requirement, it is useful for gratuitous ARPs to be flooded inside the bridge domains to update the ARP cache of other hosts.

In this example, unicasting and subnet configuration are enabled for troubleshooting purposes, such as for using `traceroute`, or for advanced features that require it, such as dynamic endpoint attachment.

**Scenario 5: Migrating to Cisco ACI, Legacy Network Connected Through a Layer 2 Extension, Gateways on Legacy Network**

In this scenario, you are migrating to Cisco ACI. You are extending Layer 2 from Cisco ACI to your legacy network, and Layer 3 gateways still reside on the legacy network.

The default gateway used by the workloads to establish communication outside of the workloads' IP subnet is initially maintained in the legacy network. This implies that the Cisco ACI fabric initially provides only Layer 2 services for devices that are part of an EPG, and the workloads that are already migrated to the Cisco ACI fabric send traffic to the legacy network when they need to communicate with devices that are external to their IP subnet.

Given the above requirements, the recommended bridge domain settings are as follows:

- L2 Unknown Unicast: Flood
  Layer 2 unknown unicast requests that originated from devices connected to the Cisco ACI fabric should be able to reach the default gateway or other endpoints that are part of the same IP subnet and are still connected to the legacy network. Because those entities are unknown to the Cisco ACI fabric, you must enable Layer 2 unknown traffic requests to flood across the Cisco ACI fabric and toward the legacy network.

- L2 Unknown Multicast Flooding: Flood
  Layer 2 unknown multicast requests that originated from devices connected to the Cisco ACI fabric should be able to reach the default gateway or other endpoints that are part of the same IP subnet and
are still connected to the legacy network. Because those entities are unknown to the Cisco ACI fabric, you must enable Layer 2 unknown traffic requests to flood across the Cisco ACI fabric and toward the legacy network.

- **Unicast Routing: Disabled**

  The Cisco ACI fabric must behave as a Layer 2 network in this initial migration phase, therefore you must disable the Unicast Routing capabilities. As a consequence, the Cisco ACI fabric will only forward traffic for endpoints that are part of this bridge domain by performing Layer 2 lookups and only MAC address information would be stored in the Cisco ACI database for those workloads (that is, their IP addresses will not be learned).

- **ARP Flooding: Enabled**

  ARP requests that originated from devices connected to the Cisco ACI fabric should be able to reach the default gateway or other endpoints that are part of the same IP subnet and are still connected to the legacy network. Because those entities are unknown to the Cisco ACI fabric, you must enable ARP requests to flood across the Cisco ACI fabric and toward the legacy network.

- **Subnet Configured: If required**

- **Enforce Subnet Check for IP Learning: If required**

In this scenario, the user is migrating hosts and services from the legacy network into the Cisco ACI fabric. A Layer 2 connection has been set up between the two environments and the Layer 3 gateway functionality will continue to exist in the legacy network for some time. The following figure illustrates the topology of this configuration:

*Figure 10: Layer 2 Connection to Fabric with External Gateways*

Afer all or the majority of the workloads belonging to the IP subnet are migrated into the Cisco ACI fabric, you can then migrate the default gateway into the Cisco ACI domain. This migration is done by turning on Cisco ACI routing in the bridge domain and disabling the default gateway function on the legacy network devices.
Cisco ACI allows you statically to configure the MAC address associated to the default gateway defined for a specific bridge domain. You can therefore use the same MAC address that you previously used for the default gateway in the legacy network so that the gateway move is completely seamless for the workloads connected to the Cisco ACI fabric. That is, there is no need to refresh the workloads' ARP cache entry.

After the migration of an application is completed, you can leverage all of the flooding containment functionalities offered by the Cisco ACI fabric. Specifically, you can disable ARP flooding as well as Layer 2 unknown unicast flooding.

This is possible only if there are no workloads belonging to that specific Layer 2 broadcast domain that remain connected to the legacy network. That is, all of the workloads, physical and virtual, have been migrated to the Cisco ACI fabric. In real life deployments, there are often specific hosts that remain connected to the legacy network for quite a long time. This is usually the case for bare-metal servers, such as Oracle RAC databases that remain untouched until the following refresh cycle. Even in this case it may make sense to move the default gateway for those physical servers to the Cisco ACI fabric. This method will provide the environment with a centralized point of management for security policies, which can be applied between IP subnets; however, the flooding of traffic must remain enabled.

After the default gateway for different IP subnets is moved to the Cisco ACI fabric, routing communication between workloads belonging to the migrated subnets will always occur on the Cisco ACI leaf nodes, leveraging the distributed anycast gateway functionality.

This is true for workloads that are still connected to the legacy network. Routing happens on the pair of border leaf nodes interconnecting legacy and new network. After workloads are migrated to the Cisco ACI fabric, traffic will be routed by leveraging the anycast gateway functionality on the leaf node where the workloads are connected.

Migrating the workloads and the workloads' default gateway to the Cisco ACI fabric brings advantages even when maintaining the security policies at the IP subnet level, as the migration allows the Cisco ACI fabric to become the single point of security policy enforcement between IP subnets, which provides a sort of ACL management functionality. You can achieve this by following a gradual procedure: after the default gateway for the different IP subnets has been moved to the Cisco ACI fabric, you can enable full and open connectivity between endpoints that are connected to different EPGs (IP subnets) by applying a "permit any" contract between the different EPGs.

With this configuration in place, every time a workload tries to communicate with a device in a different EPG (IP subnets), a centrally managed security policy is applied to the Cisco ACI leaf switch where the distributed default gateway function is enabled. Given the fact that the policy has a single "permit any" statement, this results in open connectivity between the devices.

Because routing between different IP subnets is performed at the Cisco ACI fabric level, the security policy can be enforced not only between hosts that are connected to the Cisco ACI fabric, but the security policy can also be applied to devices that are connected to VLAN segments in the legacy network.

A key advantage of the Cisco ACI centrally-managed policy system is the ability to restrict communication between hosts belonging to different IP subnets. With Cisco ACI, you can restrict communication between hosts in a holistic manner by applying a central policy from the Cisco Application Policy Infrastructure Controller (Cisco APIC), dictating which traffic flows are allowed and to and from each of the respective EPGs.
Application-Centric and Network-Centric Deployments

About Application-Centric and Network-Centric Deployments

When discussing a Cisco Application Centric Infrastructure (ACI) deployment, there are two main strategies that can be taken: application-centric and network-centric.

**Application-Centric Deployment**

When taking an application-centric approach to an ACI deployment, the applications within an organization should be allowed to define the network requirements. A true application-centric deployment will make full use of the available fabric constructs, such as endpoint groups, contracts, filters, labels, external endpoint groups, and so on, to define how applications and the tiers should communicate.

With an application-centric approach, it is generally the case that the gateways for endpoints will reside in the fabric itself (rather than on external entities such as firewalls or load balancers). This enables the application environment to get the maximum benefit from the ACI fabric.

In an application-centric deployment, much of the complexity associated with traditional networks (such as VRFs, VLANs, and subnets) is hidden from the administrator.

The following figure shows an example of an application-centric deployment:

*Figure 11: Application-Centric Deployment*

Application-centric approach is generally recommended when users fully understand their application profiles, such as the application tier and components, and what applications (or application components) need to communicate with each other and on what protocol or ports.

Application-centric deployment is also seen as an approach to on board new applications.
Benefits of using this approach include:

• Workload mobility and flexibility, with placement of computing and storage resources anywhere in the data center
• Capability to manage the fabric as a whole instead of using device-centric operations
• Capability to monitor the network as a whole using the Application Policy Infrastructure Controller (APIC) in addition to the existing operation monitoring tools; the APIC offers new monitoring and troubleshooting tools, such as health scores and atomic counters
• Lower TCO and a common network that can be shared across multiple tenants in the data center
• Reduced application downtime for network-related changes
• Rapid application deployment and agility through programmability and integrated automation
• Centralized auditing of configuration changes
• Enhanced data center security for east-west application traffic, with microsegmentation to contain threats and prevent threats from spreading laterally across tenants and applications inside the data center
• Direct visibility into the health of the application infrastructure, benefitting application owners
• Template-based configuration, which increases efficiency and enables self-service

Network-Centric Deployment

A network-centric deployment takes the opposite approach to the application-centric deployment in that the traditional network constructs, such as VLANs and VRFs, are mapped as closely as possible to the new constructs within the ACI fabric.

As an example, a traditional network deployment might consist of the following tasks:

• Define 2 server VLANs at the access and aggregation layers
• Configure the access ports to map server to VLANs
• Define a VRF at the aggregation layer
• Define an SVI for each VLAN, and map them to the VRF
• Define the HSRP parameters for each SVI
• Apply features such as ACLs to control traffic between server VLANs, and from server VLANs to the core

The comparable ACI deployment when taking a network-centric approach might be as follows:

• Deploy the fabric
• Create a tenant and VRF
• Define bridge domains for the purposes of external routing entity communication
• Create an external/inside endpoint group to communicate with external networks
• Create two bridge domains and assign a network to each indicating the gateway IP address (such as 10.10.10.1/24 and 10.10.11.1/24)
• Define the endpoint groups
• Define a “permit any” contract to allow any to any EPG communication, as a VRF would do in ‘classic’ model without ACLs

If external gateways are defined (such as firewalls or load balancers) for endpoints to use, this constitutes a network-centric approach. In this scenario, no contracts are required to allow access to the default gateway from endpoints. Although there are still benefits to be had in terms of centralized control, the fabric might become more of a Layer 2 transport in certain situations where the gateways are not inside the fabric.

The following figure shows an example of a network-centric approach:

Figure 12: Network-Centric Deployment Approach

Network-centric deployment is typically seen as a starting point for initially migrating from a legacy network to the ACI fabric, where their legacy infrastructure is segmented by VLANs, and by doing VLAN=EPG=BD mapping helps the VLANs to understand the ACI constructs better and make the transition easier.

Using this approach does not require any changes to the existing infrastructure or processes. It still can leverage the benefits that ACI offers, as listed below:

• Enables a next-generation data center network with high-speed 10- and 40-Gbps access or an aggregation network
• East-west data center traffic optimization to support virtualized, dynamic environments as well as non-virtualized workloads
• Supports workload mobility and flexibility, with placement of computing and storage resources anywhere in the data center
• Capability to manage the fabric as a whole instead of using device-centric operations
• Capability to monitor the network as a whole using the APIC in addition to the existing operation monitoring tools; the APIC offers new monitoring and troubleshooting tools, such as health scores and atomic counters

• Lower TCO and a common network that can be shared securely across multiple tenants in the data center

• Rapid network deployment and agility through programmability and integrated automation

• Centralized auditing of configuration changes

• Direct visibility into the health of the application infrastructure

Layer 2 Extension

About Layer 2 Extension

When extending a Layer 2 domain outside of the Cisco Application Centric Infrastructure (ACI) fabric to support migrations from the existing network to a new ACI fabric, or to interconnect dual ACI fabrics at Layer 2, there are the two methods to extend your Layer 2 domain:

• Extend the endpoint group out of the ACI fabric using endpoint group static path binding

• Extend the bridged domain out of the ACI fabric using an external bridged domain (also known as a Layer 2 outside)

Note

When extending the bridge domain, only a single Layer 2 outside can be created per bridge domain.

Endpoint group extension is the most popular approach to extend Layer 2 domains, where each individual endpoint group is extended using a dedicated VLAN beyond the fabric. This method is the most commonly used, as it is easy to deploy and does not require the use of contracts between the inside and outside networks.

However, if you use one bridge domain with multiple endpoint groups, then when you interconnect ACI fabrics in Layer 2, you should not use the endpoint group extension method due to the risk of loops.

Configuration Examples for Layer 2 Extension

When designing a Cisco Application Centric Infrastructure (ACI) environment for dual data centers, one topology option is to use separate fabrics, one per site, with a Layer 2 interconnection between them. In this scenario, each fabric is managed by its own Application Policy Infrastructure Controller (APIC) cluster, with no sharing or synchronization of policies between each.

The following figure illustrates interconnecting ACI fabrics at Layer 2:
In this example, multiple endpoint groups are associated with a single bridge domain. In this scenario, you should not extend each individual endpoint group between fabrics as shown in the figure, as this might result in loops between the fabrics. Instead, a Layer 2 Outside should be used to extend the entire bridge domain using a single VLAN, as shown in the following figure:

**Figure 14: Interconnect Fabrics at Layer 2 - Multiple Endpoint Groups per Bridge Domain (Recommended Scenario)**

---

**Additional References for Layer 2 Extension**

For more information about Layer 2 extension, see the "ACI Layer 2 Connection to the Outside Network" section of the Connecting Application Centric Infrastructure (ACI) to Outside Layer 2 and 3 Networks white paper at the following URL:

Infrastructure VXLAN Tunnel Endpoint Pool

About Infrastructure VXLAN Tunnel Endpoint Pool

The Cisco Application Centric Infrastructure (ACI) fabric is brought up in a cascading manner, starting with the leaf nodes that are directly attached to the Application Policy Infrastructure Controller (APIC). LLDP and control-plane IS-IS convergence occurs in parallel to this boot process. The ACI fabric uses LLDP- and DHCP-based fabric discovery to automatically discover the fabric switch nodes, assign the infrastructure VXLAN tunnel endpoint (VTEP) addresses, and install the firmware on the switches.

The VTEP pool, which is specified only once during fabric discovery, is the pool of addresses used while building the fabric. That is, each switch node added to the fabric gets an address. The VTEP pool is used for other infrastructure related extensions, such as extending the infrastructure into a host for Application Virtual Switch (AVS) integration.

Prerequisites for Infrastructure VXLAN Tunnel Endpoint Pool

You must meet the following prerequisites to use infrastructure VXLAN Tunnel Endpoint Pool (VTEP):

- The Application Policy Infrastructure Controllers (APICs) are clean and have no configuration. The only time the VTEP pool gets set for the infrastructure is during the startup script on the APICs.
- The leaf and spine nodes to be added to the fabric are running a Cisco Application Centric Infrastructure (ACI) image and not an NX-OS standalone image.
- The leaf and spine nodes to be added to the fabric are not part of another ACI fabric.

Guidelines and Limitations for Infrastructure VXLAN Tunnel Endpoint Pool

The following guidelines and limitations apply for infrastructure VXLAN Tunnel Endpoint Pool (VTEP):

- The infrastructure VTEP address cannot be changed once the fabric is built around it.
- To change the VTEP pool, the fabric must be rebuilt from scratch. This is a disruptive process and will require the configuration to be exported, then imported after the initial fabric steps are completed.
- Generally, the infrastructure subnet stays internal to the fabric. The subnet exists within its own VRF and is rarely exposed beyond that.
- There are a few scenarios, such as Application Virtual Switch (AVS) integration, where this subnet gets exposed outside of the fabric. Due to this, ensure that this subnet does not overlap with another subnet that is in use within the data center.
- While the minimum supported subnet size is a /22, this is not an ideal pool size and will cause scale issues while attempting to grow the fabric. Subnet size /22 is only recommended for a small lab environment.

If subnet size is a concern, a recommended subnet size for the VTEP pool is a /19. Otherwise, the ideal subnet size for the VTEP pool is a /16.
Recommended Configuration Procedure for Infrastructure VXLAN Tunnel Endpoint Pool

The Infrastructure VTEP pool only ever gets set on the Application Policy Infrastructure Controller (APIC) during the startup script before the fabric is ever built.

Verifying the Infrastructure VXLAN Tunnel Endpoint Pool

The point at which the infrastructure VTEP pool can be verified is right before accepting the configuration within the startup script on the Application Policy Infrastructure Controller (APIC). The APIC asks if the configuration is correct, including the VTEP pool address assignment. After you confirm that the configuration is correct, the larger pool gets broken into multiple DHCP pools for various purposes within the fabric and there is currently no straightforward way to verify the initial pool size after startup script acceptance.

That being said, with the APIC connected to the fabric, the following procedure can be used to observe the pools that the initial TEP pool was carved up into, and subsequently the initial network it is carved from.

Procedure

Use the `moquery -c dhcpPool` command to view the TEP pool configuration.

Example:

```
Apic1# moquery -c dhcpPool
...  
  dn : prov-3/net-[10.0.0.0/16]/pool-7
```

Specifically within the output distinguished name of this class, there is a section that begins with "net-". In the example snippet above, the APIC was configured with 10.0.0.0/16 as its TEP pool within the setup script of the APIC.

Configuration Examples for Infrastructure VXLAN Tunnel Endpoint Pool

The default configuration is 10.0.0.0/16. The configuration is only set once during the startup script on the Application Policy Infrastructure Controller (APIC).

Additional References for Infrastructure VXLAN Tunnel Endpoint Pool

For more information on setting up the Application Policy Infrastructure Controller (APIC), see the *Cisco APIC Getting Started Guide* at the following URL:

Virtual Routing and Forwarding Instances

About Virtual Routing and Forwarding Instances

A virtual routing and forwarding (VRF) instance, also called a context, represents an application policy domain and Layer 3 forwarding. A tenant can have one or more VRF instances, and a single VRF instance can have one or more bridge domains. A VRF instance in Cisco Application Centric Infrastructure (ACI) is equivalent to a VRF instance in a traditional network.

Guidelines and Limitations for Virtual Routing and Forwarding Instances

The following guidelines and limitations apply for virtual routing and forwarding (VRF) instances:

• Within a single VRF instance, IP addresses must be unique. Between different VRF instances, you can have overlapping IP addresses.

• If shared services is used between VRF instances or tenants, make sure there are no overlapping IP addresses.

• Any VRF instances that are created in common tenant will be seen in other user-configured tenants.

• VRF supports enforced mode or unenforced mode. By default, a VRF instance is in enforced mode, which means all endpoint groups within the same VRF instance cannot communicate to each other unless there is a contract in place.

• Switching from enforced to unenforced mode (or vice versa) is disruptive.

Additional References for Virtual Routing and Forwarding Instances

For more information about virtual routing and forwarding (VRF) instances, see the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:


Stretched Fabric

About Stretched Fabric

The stretched fabric allows users to manage multiple datacenter sites as a single fabric by using the same Application Policy Infrastructure Controller (APIC) controller cluster. The stretched Cisco Application Centric Infrastructure (ACI) fabric behaves the same way as a regular ACI fabric to support workload portability and virtual machine mobility. The following figure illustrates the stretched fabric topology:
Guidelines and Limitations for Stretched Fabric

The following guidelines and limitations apply for stretched fabric:

- Cisco Application Centric Infrastructure (ACI) stretched fabric site-to-site connectivity options include dark fiber, dense wavelength division multiplexing (DWDM), and Ethernet over MPLS (EoMPLS) pseudowire.

- The current validated stretched fabric supports three sites.

- The maximum validated/supported distance between two sites is up to 800 KM/500 miles or latency within 10 msec RTT to allow Application Policy Infrastructure Controller (APIC) controller clusters to keep control and data synchronized.

- With software release 1.2(2g), the ACI fabric supports up to six MP-BGP route reflectors. In a stretched fabric implementation with three sites, place two route reflectors at each site to provide redundancy.

- Transit leaf refers to the leaf switches that provide connectivity among sites. There are no special requirements and no additional configurations required for transit leaf switches.

- Transit leaf switches in all sites connect to both the local and remote spine switches.

- One or more transit leaf switches can be used. The number of transit leaf switches and links are dictated by redundancy and bandwidth capacity decisions.

- In the event of link failure between sites, bring the failed links back up so as to avoid system performance degradation, or to prevent a split fabric scenario from developing.

- Bridge domains/IP subnets can be stretched between sites.
Additional References for Stretched Fabric

For more information about stretched fabric, including failure scenarios and more operational guidelines, see the ACI Stretched Fabric Design knowledge base article at the following URL:


Access Policies

About Access Policies

The Fabric tab in the Cisco Application Policy Infrastructure Controller (APIC) GUI is used to configure system-level features including, but not limited to, device discovery and inventory management, diagnostic tools, domain configuration, and switch and port behavior. The fabric pane is split into three sections: Inventory, Fabric Policies, and Access Policies. Understanding how fabric and access policies configure the fabric is key for maintaining these policies for the purposes of internal connections between fabric leaf nodes, connections to external entities such as servers, networking equipment, and storage arrays.

This section lists guidelines and provides common configuration examples for key objects in the Fabric > Access Policies view. The Access Policies view is split into folders separating out different types of policies and objects that affect fabric behavior. For example, the Interface Policies folder is where port behavior is configured such as port speed and the controls for specifying whether or not to run protocols, such as LACP, on switch interfaces. Domains and AEPs are also created in the Access Policies view. The fabric access policies provide the fabric with the base configuration of the access ports on the leaf switches. For more information, see Additional References for Access Policies, on page 42.

Guidelines and Limitations for Access Policies

Cisco has established several best practices for fabric configuration. These are not requirements, and might not work for all environments or applications, but might help simplify day-to-day operation of the Cisco Application Centric Infrastructure (ACI) fabric.

This section contains basic guidelines for access policies.

General Guidelines

• Policies should be created once and reused when connecting new devices to the fabric. Maximizing the reusability of policy and objects makes day-to-day operations exponentially faster and easier to make large-scale changes.

Note

The usage of these policies can be viewed by clicking the Show Usage button in the Application Policy Infrastructure Controller (APIC) GUI. Use this to determine what objects are using a certain policy to understand the impact when making changes.

• Avoid using the Basic GUI or Quick Start wizards, as these may create many automatic configurations that are not intuitive during troubleshooting.
Interface Policy Guidelines

- Do not use the default setting for interface policies, if possible.
- Reuse policies whenever possible. For example, create new separate interface policies for LACP active, passive, and mac-pinning; for 1-GE port speed and 10-GE port speed; and for CDP and LLDP policies.
- When naming interface policies, use names that clearly describe the setting. For example, a policy that enables LACP in mode active could be called "LACP-Active". There are many default policies out of the box. However, it can be hard to remember what all the defaults are, which is why policies should be clearly named to avoid making a mistake when adding new devices to the fabric.

Domain Guidelines

- Build one physical domain per tenant for bare metal servers or servers without hypervisor integration requiring similar treatment.
- Build one external routed/bridged domain per tenant for external connectivity.
- For VMM domains, if both DVS and AVS is in use, create a separate VMM domain to support each environment.
- For large deployments where domains (physical/VMM/etc) need to be leveraged across multiple tenants, a single physical domain or VMM domain can be created and associated with all leaf ports where services are connected.

AEP Guidelines

- Multiple domains can be associated to a single AEP for simplicity. There are some cases where multiple AEPs may need to be configured to enable the infrastructure VLAN, such as overlapping VLAN pools, or to limit the scope of the presence of VLANs across the fabric.
- Another scenario in which multiple AEPs should be utilized is when making an association to VMM domains. The AAEIP also contains relationships to the vSwitch policies, which are then pushed to the vCenter VDS or AVS. If there are multiple VMM domains deployed with differing vSwitch policies, multiple AAEIPs should be created to account for the various potential vSwitch policy combinations.
- When utilizing an AVS for VMM, HyperV, SCVMM, or OpenStack OpFlex integration, the AAEIP is where the option to enable infra vlan is selected. For the most part, we do not want to extend this VLAN outside of the fabric aside for when performing this integration. For that purpose, it will be beneficial to create an AEP specific to the AVS VMM Domain if being utilized.

Configuration Examples for Access Policies

This section describes two common methods for deploying your leaf switches, explains how to create and associate switch and interface profiles, and shows how to create a port channel policy and a vPC domain.

Creating Access Policies for Switches

One common method for deploying your leaf switches is to create a switch profile for each leaf switch individually. Additionally, create a switch profile for each vPC pair (if you are using vPC).

You also create an interface profile for each switch profile. Each interface profile will group all the interface selectors associated to that specific switch. In the event of adding or deleting new or existing ports, changes...
will only be made under interface profiles, as those interface profiles are already associated to the corresponding switch profiles.

Consider the following vPC topology as an example:

![vPC Topology Diagram](image)

- **When a switch profile is created for each leaf switch individually regardless of vPC definitions:**
  - Switch profiles example: Leaf_201, Leaf_202
  - Interface profiles example: Leaf_201_IPR, Leaf_202_IPR

  In the example above, all ports (vPC or non-vPC) are added in both Leaf_201_IPR and Leaf_202_IPR respectively.

  The benefits of creating a switch profile for each leaf individually regardless of vPC definitions are that there are less switch and interface profiles to manage, it's more flexible to change the ports if needed, and it supports asymmetric connections for host-facing ports. However, the interface policy group needs to be configured consistently on both interface selectors.

- **When a switch profile is created for each leaf switch individually and also for each vPC pair:**
  - Switch profiles example: Leaf_201, Leaf_202, Leaf_201_202
  - Interface profiles example: Leaf_201_IPR, Leaf_202_IPR, Leaf_201_202_IPR

  In the example above, vPC related ports are only added in Leaf_201_202_IPR. Non-vPC related ports are added to either Leaf_201_IPR or Leaf_202_IPR respectively.

  The benefit of creating a switch profile for each leaf switch and also for each vPC pair is that the configurations are simpler in a large-scale environment with symmetric in discipline and replicated setup. However, it is difficult to repurpose the ports that are already in use. Changing those interfaces will impact both of the switches.

This section explains how to create and associate switch and interface profiles.

**Creating a Switch Profile**

This section explains how to create a switch profile (leaf or spine).

**Before you begin**

You must have a configured leaf or spine switch.
Creating an Interface Profile

This section explains how to create a switch profile (leaf or spine).

Before you begin
You must have a configured leaf or spine switch.

Procedure

Step 1 From the Fabric tab, click Access Policies.
Step 2 In the Navigation pane, choose Switch Policies > Profiles. The Leaf Profile and Spine Profile options appear in the Navigation pane.
Step 3 Choose Leaf Profile or Spine Profile.
Step 4 In the Work pane, click Actions and choose the option to create a profile. A dialog appears. When creating a leaf profile, the Create Leaf Profile dialog appears. When creating a spine profile, the Create Spine Profile dialog appears.
Step 5 Enter the appropriate values in the fields of the dialog.
Note For an explanation of a field, click the 'i' icon on the top-right corner of the dialog box to display the help file.
Step 6 When done, click Submit.

Associating Switch and Interface Profiles

Before you begin
• You have created a switch (leaf or spine) profile.
• You have created an interface (leaf or spine) profile.
This section explains how to associate switch profiles with interface profiles.

**Procedure**

**Step 1**  
From the Fabric tab, click Access Policies.

**Step 2**  
In the Navigation pane, click Switch Policies > Profiles.  
The Leaf Profile and Spine Profile options appear in the Navigation pane.

**Step 3**  
Click the Leaf Profile or Spine Profile drop-down arrow.  
Your profile icons appear in the drop-down list in the Navigation pane.

**Step 4**  
In the Navigation pane, click on a profile icon to choose a switch profile.  
Your profile details appear in the Work pane.

**Step 5**  
From the Associated Interface Selector Profiles table in the Work pane, click the + (plus) symbol.  
The Create Interface Profile dialog appears.

**Step 6**  
Click the Interface Select Profile drop-down arrow and choose an interface profile to associate with your switch profile.

**Step 7**  
When done, click Submit.

---

### Creating a Port Channel Policy

This section explains how to create a port channel policy.

**Procedure**

**Step 1**  
From the Fabric tab, click Access Policies.

**Step 2**  
In the Navigation pane, choose Interface Policies > Policies > Port Channel.

**Step 3**  
From the Work pane, click Actions > Create Port Channel Policy.  
The Specify Port Channel Policy dialog appears.

**Step 4**  
Enter the appropriate values in the Specify Port Channel Policy dialog fields.

**Note**  
- For an explanation of a field, click the 'i' icon on the top-right corner of the dialog box to display the help file.
- The LACP Active option for the Mode field sets a port to the suspended state if it does not receive an LACP PDU from the peer. Although this feature helps in preventing loops created due to misconfigurations, in some cases, the feature can cause servers to fail to boot up because they require LACP to logically bring up the port. This is the use case that you typically would see with PXE boot. As a workaround, you click the checked Suspend-Individual Port check box in the Control options to uncheck/disable the option and put a port into an individual state.

**Step 5**  
When finished, click Submit.

---

### Creating a vPC Domain

For server active/active deployments, vPC can be used to provide larger uplink bandwidth and faster convergence upon link or switch failures.
Unlike traditional vPC design, there is no requirement for setting up either a vPC peer-link or vPC peer-keepalive in the Cisco Application Centric Infrastructure (ACI) fabric. The fabric itself serves as the peer-link. The rich interconnectivity between spine switches and leaf switches makes it very unlikely that all the redundant paths between vPC peers fail at the same time. Hence, if the peer switch becomes unreachable, it is assumed to have crashed. The slave switch does not bring down vPC links.


**Procedure**

**Step 1**  
From the Fabric tab, click Access Policies.

**Step 2**  
In the Navigation pane, click Switch Policies > Policies > Virtual Port Channel default. The Virtual Port Channel Security Policy - Virtual Port Channel default window appears.

**Step 3**  
Enter the appropriate values in the fields of the Virtual Port Channel Security Policy - Virtual Port Channel default window.

**Note**  
For an explanation of a field, click the 'i' icon on the top-right corner of the dialog box to display the help file.

**Step 4**  
When finished, click Submit.

---

### Additional References for Access Policies


### Mis-Cabling Protocol

#### About the Mis-Cabling Protocol

Unlike traditional networks, the Cisco Application Centric Infrastructure (ACI) fabric does not participate in the Spanning Tree Protocol (STP) and does not generate bridge protocol data units (BPDUs). BPDUs are instead transparently forwarded through the fabric between ports mapped to the same endpoint group. Therefore, Cisco ACI relies to a certain degree on the loop prevention capabilities of external devices.

Some scenarios, such as the accidental cabling of two leaf ports together, are handled directly using LLDP in the fabric. However, there are some situations where an additional level of protection is necessary; in those cases, enabling the Mis-Cabling Protocol (MCP) can help.

Consider the example in the following figure:
In this example, two endpoint groups are configured on the Cisco ACI fabric, both associated with the same bridge domain. An external switch has one port connected to each of the endpoint groups. In this example, a misconfiguration has occurred whereby the external switch is allowing VLAN 10 on port 1/20; however, the endpoint group associated with port 1/10 on leaf 102 is configured for VLAN 11. In this case, port 1/10 on leaf 102 will not be able to receive BPDUs for VLAN 10. As a result, the spanning tree cannot detect the loop and all ports will be forwarding.

The MCP protocol, if enabled, provides additional protection against this type of misconfiguration. MCP is a lightweight protocol designed to protect against loops that cannot be discovered by either STP or LLDP. You should enable MCP on all ports facing external switches or similar devices.

Per-VLAN MCP will only run on 256 VLANs per interface. If there are more than 256 VLANs, then the first numerical 256 VLANs are chosen.

Configuration Examples for the Mis-Cabling Protocol

To enable the Mis-Cabling Protocol (MCP) in the fabric, you must enable MCP globally through the global policies and also on individual ports or port channels through the interface policy group configuration.

Procedure

Step 1  On the menu bar, choose Fabric > Access Policies.
Step 2  In the Navigation pane, choose Global Policies > MCP Insurance Policy Default.
Step 3  In the Work pane, for the Admin State buttons, choose Enabled.
Step 4  For the remaining properties, change the values as desired.

- **Key** and **Confirm Key**—A key that uniquely identifies MCP packets within the fabric.
- **Initial Delay (sec)**—The delay time in seconds before MCP begins taking action.
- **Loop Detect Multiplication Factor**—Denotes the number of continuous packets a port must receive before declaring a loop.
Step 5  Enable MCP on the interface level, which is done when you create an access port policy group. On the menu bar, choose Fabric > Access Policies.
Step 6  In the Navigation pane, choose Interface Policies > Policy Groups.
Step 7  In the Work pane, choose Actions > Create Access Policy Group.
Step 8  In the Create Access Policy Group dialog box, in the MCP Policy drop-down list, choose MCP-Enabled.
Step 9  Fill out the remaining fields as necessary.
Step 10 Click Submit.

Additional References for the Mis-Cabling Protocol

For more information about the Mis-Cabling Protocol (MCP), see the section about loop detection in the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:


Port Tracking

About Port Tracking

Port tracking policies are used to monitor the status of links between leaf switches and spine switches. When an enabled port tracking policy is triggered, the leaf switches take down all access interfaces on the switch that have endpoint groups deployed on them.

Port tracking addresses a scenario in which a leaf node might lose connectivity to the spine node and where hosts connected to the affected leaf node in an active/standby manner might not be aware of the failure for a period of time. The following figure illustrates this scenario:

The port tracking feature detects a loss of fabric connectivity on a leaf node and brings down the host facing ports. This allows the host to fail over to the second link, as shown in the following figure:
The preferred host connectivity to the Cisco Application Centric Infrastructure (ACI) fabric is vPC wherever possible. Port tracking is useful in situations where hosts are connected using active/standby NIC teaming.

Guidelines and Limitations for Port Tracking

- The preferred host connectivity to the ACI fabric is vPC wherever possible.
- Port tracking is useful in situations where hosts are connected using active/standby NIC teaming.

Recommended Configuration Procedure for Port Tracking

To enable and set global port tracking for the ACI fabric, complete the following steps.

Procedure

Step 1
In the Advanced GUI, navigate to the Port Tracking window. Click Fabric > Access Policies > Global Policies > Port Tracking.

Step 2
In the Port Tracking window, locate the Port Tracking state field and click on.

Step 3
Set the Delay restore timer parameter. This timer controls the number of seconds the fabric waits before bringing host ports up after the leaf spine links re-converge.

Step 4
Set the Number of Active Spine Links parameter. This value specifies how low the number of active links drop to before Port Tracking is triggered. The value '0' configures Port Tracking to be triggered after the number of active links to the spine drops to zero.

Step 5
Click Submit.
VLAN Pools

About VLAN Pools

Within Cisco Application Centric Infrastructure (ACI), there is the concept of access policies, which are a group of objects that define how traffic can get access into the fabric. Access policy definition matters when an EPG is created for use. For example, an EPG that has a static path (for example, node 101, int eth1/10, trunked with VLAN 10) without access policies is essentially telling the EPG to use a set of policies to which it does not have access. At this point, you will see faults indicating path issues. The access policies and subsequent domain-to-EPG association tell this EPG that it now has access to a subset of nodes, interfaces, and VLANs that it can now use in path definitions.

VLAN pools are just one piece of the complete access policies definition. A VLAN pool is a container that is comprised of encap blocks, which contain the actual VLAN definitions.

Prerequisites for VLAN Pools

- A Cisco ACI fabric that has been initialized.
- An understanding of access policies and their purpose. For information on access policies, see About Access Policies, on page 37

Guidelines and Limitations for VLAN Pools

- VLAN pools containing overlapping encap block definitions should not be associated to the same AAEP (and subsequently the same leaf nodes). This can cause issues with BPDU forwarding through the fabric if the domains associated to an EPG have overlapping VLAN block definitions.
- VLAN pools with an allocation mode of Dynamic are typically used for VMM integration deployments. VMM integration generally does not require explicit VLAN assignment, so a dynamic pool allows the system to pull free resources as needed.
- VLAN pools with an allocation mode is Static are typical for the majority of other deployment scenarios including static paths, L2Out and L3Out out definitions.
- A dynamic VLAN pool can have a static encap block defined within it. This is generally only done for the specific case of utilizing the "pre-provision" resolution immediacy.
- A static VLAN pool cannot have a dynamic encap block. This will be rejected by the Application Policy Infrastructure Controller (APIC), as there are no features that utilize this configuration.

Recommended Configuration Procedures for VLAN Pools

See Guidelines and Limitations for VLAN Pools.

Configuration Examples for VLAN Pools

For configuration examples of VLAN pools, please see the Creating Domains, Attach Entity Profiles, and VLANs to Deploy an EPG on a Specific Port document at the following URL:

Additional References for VLAN Pools

For additional information on access policies, including VLAN pools, see the *Cisco Application Centric Infrastructure Fundamentals* document at the following URL:


Managed Object Naming Convention

About the Managed Object Naming Convention

Cisco Application Centric Infrastructure (ACI) is based upon the managed object (MO) model, where each object requires a name. A clear and consistent naming convention is therefore essential to aid manageability and troubleshooting.

Any change in naming convention for any MO such as profiles or policies requires disruption. It is highly recommended to plan ahead and define the policy naming convention before deploying the ACI fabric to ensure that all policies are named consistently.
About the Managed Object Naming Convention
Routing Design

- Transit Routing, on page 49
- L3Out Ingress Policy Enforcement, on page 64
- L3Out MTU Considerations, on page 67
- Shared L3Outs, on page 69
- L3Out Router IDs, on page 73
- Multiple External Connectivity, on page 77

Transit Routing

About Transit Routing

The Cisco Application Centric Infrastructure (ACI) solution allows you to use standard Layer 3 technologies to connect to external networks. These can be Layer 3 connections to an existing network, WAN routers, firewalls, mainframes, or any other Layer 3 device. Border leaf switches within the Cisco ACI fabric provide connectivity to the external Layer 3 devices. Cisco ACI supports Layer 3 connections using static routing (IPv4 and IPv6) or the following dynamic routing protocols:

- OSPFv2 (IPv4) and OSPFv3 (IPv6)
- BGP (IPv4 and IPv6)
- EIGRP (IPv4 and IPv6)

Within the Cisco ACI fabric, multiprotocol BGP (MP-BGP) is implemented between the leaf and spine switches to propagate external routes within the fabric. The BGP route reflector technology is deployed to support many leaf switches within a single fabric. All of the leaf and spine switches are in one single BGP autonomous system (AS). Once the border leaf learns the external routes, it can then redistribute the external routes of a given VRF instance to an MP-BGP address family (VPNv4 or VPNv6). MP-BGP maintains a separate BGP routing table for each VRF instance. Within MP-BGP, the border leaf switch advertises routes to a spine switch, which is a BGP route reflector. The routes are then propagated to all the leaf switches where the VRF instances are instantiated.

Before Cisco Application Policy Infrastructure Controller (Cisco APIC) release 2.3(1f), transit routing was not supported within a single L3Out profile. In Cisco APIC release 2.3(1f) and later, you can configure transit routing with a single L3Out profile, with the following limitations:
• If the VRF instance is unenforced, an external subnet (l3extSubnet) of 0.0.0.0/0 can be used to allow traffic between the routers sharing the same Layer 3 EPG.

• If the VRF instance is enforced, an external default subnet (0.0.0.0/0) cannot be used to match both source and destination prefixes for traffic within the same Layer 3 EPG. To match all traffic within the same Layer 3 EPG, the following prefixes are supported:
  
  • IPv4
    • 0.0.0.0/1—with External Subnets for the External EPG
    • 128.0.0.0/1—with External Subnets for the External EPG
    • 0.0.0.0/0—with Import Route Control Subnet, Aggregate Import
  
  • IPv6
    • 0::0/1—with External Subnets for the External EPG
    • 8000::0/1—with External Subnets for the External EPG
    • ::0/0—with Import Route Control Subnet, Aggregate Import

• Alternatively, a single default subnet (0.0.0.0/0) can be used when combined with a VzAny contract. For example:
  
  • Use a VzAny providing contract and a Layer 3 EPG consuming contract (matching 0.0.0.0/0), or a VzAny consuming contract and Layer 3 EPG providing contract (matching 0.0.0.0/0).
  
  • Use the subnet 0.0.0.0/0—with Import/Export Route Control Subnet, Aggregate Import, and Aggregate Export.

Figure 17: Multiprotocol BGP Transit Peering Topology
Prerequisites for Transit Routing

To configure transit routing, you must meet the following prerequisites:

• You must have configured multiple external Layer 3 connections within the same VRF

• You must have configured a BGP route reflector policy for Cisco Application Centric Infrastructure (ACI) fabric

Guidelines and Limitations for Transit Routing

Transit routing is not enabled as a feature itself, but is allowed when export policies are configured that allow external routes from one external Layer 3 connection to be advertised out another external Layer 3 connection. External Layer 3 connections are configured using the Layer 3 Outside object in Cisco Application Centric Infrastructure (ACI). Layer 3 Outside connections (commonly referred to as L3Outs) are supported for the following connection types:

• Static connection
• OSPF (all area types)
• EIGRP
• iBGP over direct connection
• iBGP over OSPF (iBGP multihop)
• iBGP over static route (iBGP multihop)
• eBGP over direct connection
• eBGP over OSPF (eBGP multihop)

Not all transit routing combinations are currently supported in ACI. For information about the currently supported transit routing combinations, see the Cisco APIC and Transit Routing document at the following URL:


Recommended Configuration Procedure Transit Routing

External Layer 3 connectivity is configured in Cisco Application Centric Infrastructure (Cisco ACI) using the Layer 3 Outside configuration policy (commonly referred to as L3Out). Cisco ACI supports multiple L3Out connections per tenant and VRF instance. When multiple L3Outs are configured in the same tenant and VRF instance, external routes learned from one L3Out can be advertised through another L3Out, making the Cisco ACI fabric a transit network. The propagation of externally learned routes from one L3Out to another L3Out is controlled by a policy with the default behavior to not advertise externally learned routes from one L3Out to another L3Out.

L3Outs are deployed on Cisco ACI leaf switches. When an L3Out is configured on a leaf switch, this effectively makes the leaf switch a border leaf switch. Multiple border leaf switches can be configured in each tenant and VRF instance.
From a routing perspective, the Cisco ACI fabric does not function as a single logical router, but rather as a network of routers that are connected to an MP-BGP core. All routes learned from an L3Out are leaked into MP-BGP and then redistributed to every leaf switch in the fabric where the VRF instance is deployed. If another L3Out is configured on another leaf switch, those routes can be advertised back out the other L3Out. This provides transit routing functionality to the Cisco ACI fabric. Transit routing is supported on the same leaf switch or on different leaf switches and is supported for a number of different combinations, such as OSPF to OSPF, BGP to OSPF, and EIGRP to static.

**OSPF to OSPF Transit on Different Leaf Switches**

When multiple OSPF L3Outs are configured on different leaf switches in the same VRF instance, the OSPF areas are in separate OSPF domains. For example, the same area deployed on different switches will not be joined together. When routes from one OSPF L3Out are permitted out of the other OSPF L3Out, the routes will appear as OSPF external type-2 routes (Type-5 LSAs).

*Figure 18: OSPF to OSPF Transit on Different Leaf Switches*

Both L3Outs are configured in the same VRF instance and use the same OSPF area ID, but are in different OSPF domains. Routes learned on border leaf switch 1 in OSPF area 10 will appear as OSPF learned routes on border leaf switch 1. These routes will appear as BGP learned routes on all other leaf switches in the fabric where VRF1 is instantiated, including border leaf switch 2. The following output shows the OSPF learned routes received on border leaf switch 1:
The bolded line is the external route that is learned from the L3Out (OSPF).

The following output shows the same route learned on border leaf switch 2, in which the route is learned through MP-BPG:

```
BL-2# show ip route 10.100.100.0/24 vrf prod:ctx1
IP Route Table for VRF "prod:ctx1"
'***' denotes best mcast next-hop
'[x/y]' denotes [preferences/metric]
'@<string>' in via output denotes VRF <string>
10.100.100.0/24, ubest/mbest: 1/0
  *via 10.0.112.95%overlay-1, [200/5], 00:30:04, bgp-65000, internal, tag 65000
  recursive next hop: 10.0.112.95/32%overlay-1
```

The bolded lines are the route that is learned from the fabric (MP-BPG).

By default, Cisco ACI will not advertise routes learned from one L3Out back out another L3Out. The Cisco ACI does not allow transit by default. Transit routing is controlled by creating export route control policies for the L3Out. Export route control policies control which transit prefixes are redistributed into the L3Out protocol. These policies will be instantiated on the leaf switch as route maps and IP prefix-lists.

By looking at the OSPF process information on border leaf switch 2, you can see how this policy is instantiated on border leaf switch 2 using redistribution with route-maps and IP prefix-lists:

```
BL-2# show ip ospf vrf prod:ctx1
Routing Process default with ID 1.1.1.103 VRF prod:ctx1
Stateful High Availability enabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
Table-map using route-map exp-ctx-3047429-deny-external-tag
Redistributing External Routes from
  static route-map exp-ctx-st-3047429
direct route-map exp-ctx-st-3047429
bgp route-map exp-ctx-proto-3047429
eigrp route-map exp-ctx-proto-3047429
```

The bolded lines of output show the redistribution of external routes from BGP and EIGRP.

```
BL-2# show route-map exp-ctx-st-3047429
route-map exp-ctx-st-3047429, permit, sequence 7801
  Match clauses:
    ip address prefix-lists: IPv6-deny-all IPv4/proto32771-3047429-exc-ext-inferred-export-dst
  Set clauses:
    tag 4294967295
```

The output shows the route map.

```
BL-2# show ip prefix-list IPv4/proto32771-3047429-exc-ext-inferred-export-dst
ip prefix-list IPv4/proto32771-3047429-exc-ext-inferred-export-dst: 1 entries
  seq 1 permit 10.100.100.0/24
```

The output shows the IP prefix-list.
The OSPF database on border leaf switch 2 shows that the prefix 10.100.100.0/24 is learned by redistribution into OSPF and not as an intra-area prefix. Both OSPF L3Outs that are being deployed on different border leaf switches use the same area ID, but are in different OSPF domains. Each border leaf switch is an ASBR that redistributes fabric learned prefixes into the OSPF process that is local to that leaf switch.

```
BL-2# show ip ospf database 10.100.100.0 vrf prod:ctx1
OSPF Router with ID (1.1.1.103) (Process ID default VRF prod:ctx1)

Type-5 AS External Link States

Link ID    ADV Router    Age   Seq#       Checksum    Tag
10.100.100.0 1.1.1.103  494  0xb000002  0xdeb4  4294967295
```

The output shows that the route L3Out on border leaf switch 1 is added as a type 5 external LSA on border leaf switch 2.

The following figure shows the same topology from a routing protocol view:
The border leaf switches run both BGP (within the fabric) and OSPF for external connectivity. The mutual redistribution is done on the border leaf switches.

**OSPF to OSPF Transit on the Same Border Leaf Switch**

OSPF L3Outs can be deployed on the same border leaf switches. In this case, the transit routes will be local to the switch. When using OSPF L3Outs, you have the following options depending on the topology:

- All external devices connect to a border leaf switch are in the different OSPF areas—Use different OSPF L3Outs.
- All external devices connected to a border leaf switch are in the same OSPF area—Use a single OSPF L3Out with multiple interfaces.
Before Cisco APIC, release 2.3(1f), transit routing was not supported within a single L3Out profile. In Cisco APIC, release 2.3(1f) and later, you can configure transit routing within a single L3Out profile, with limitations; for details, see About Transit Routing, on page 49.

Different OSPF Areas Connected to the Same Border Leaf Switch

When a border leaf switch is connected to multiple OSPF areas, the border leaf switch will become an OSPF area border router (ABR). The OSPF rules for an ABR state that one area must be connected to area 0; OSPF virtual links are not supported in Cisco ACI. This rule holds true for an Cisco ACI border leaf switch. When an Cisco ACI border leaf connects to multiple OSPF areas in the same VRF instance, one area should be in area 0. This is required to support transit routing between the areas.

An L3Out can only belong to one area; therefore, when connecting to different OSPF areas, different L3Outs must be used. Cisco ACI still blocks transit routes between different L3Outs unless permitted by a policy, but
instantiation of this policy is different for an ABR. Cisco ACI blocks transit routes between different prefixes using an OSPF area filter-list. The OSPF filter-list blocks OSPF type-3 LSAs.

**Note**

The area filter-list implementation only filters type-3 LSAs. If external type-5 or type-7 (NSSA) LSAs are learned from an OSPF L3Out on the ABR, these routes will be permitted to other areas connected to the ABR.

*Figure 21: Area Filter-List Filtering Type-3 LSAs Between Areas*

When export route control subnets are added to the L3Out, the IP prefix-list for the subnet will be added to the route-map used for the filter-list as well as the redistribute command.

```
Number of active areas is 2, 2 normal, 0 stub, 0 nssa
Area (00.0.0.10) (Inactive)
  Area has existed for 00:28:57
  Interfaces in this area: 2 Active interfaces: 2
  Passive interfaces: 1 Loopback interfaces: 1
  SPF calculation has run 11 times
  Last SPF ran for 0.000117s
  Area ranges are
  **Area-filter in 'exp-ctx-proto-2949124'**
  Number of LSAs: 3, checksum sum 0x0
Area (backbone)
  Area has existed for 03:14:11
  Interfaces in this area: 2 Active interfaces: 1
  Passive interfaces: 1 Loopback interfaces: 1
  SPF calculation has run 21 times
  Last SPF ran for 0.000234s
  Area ranges are
```
Area-filter in 'exp-ctx-proto-2949124'
Number of LSAs: 2, checksum sum 0x0

The bolded lines are the route-maps that are used with the OSPF area filter.

```
BL-1# show route-map exp-ctx-st-2949124'
route-map exp-ctx-st-2949124', permit, sequence 7801
    Match clauses:
        ip address prefix-lists: IPv6-deny-all IPv4-proto49155-2949124-exc-ext-inferred-export-dst
    Set clauses:
        tag 4294967295

Leaf-3# show ip prefix-list IPv4-proto49155-2949124-exc-ext-inferred-export-dst
ip prefix-list IPv4-proto49155-2949124-exc-ext-inferred-export-dst: 1 entries
    seq 1 permit 10.1.1.0/24
```

**Note**

When multiple OSPF L3Outs are configured on the same border leaf switch, they are configured under the same OSPF process. Export route control subnets and public bridge domain and endpoint group subnets are added to route-maps used by redistribution into OSPF. When a subnet is allowed out one OSPF L3Out on the border leaf switch, it will apply to all OSPF L3Outs on the same border leaf switch. This is also true for multiple EIGRP L3Outs on the same border leaf switch.

**Same OSPF Area Connected to the Same Border Leaf Switch**

When connecting to multiple external devices from the same border leaf switch that are in the same OSPF area, only one L3Out is used. L3Outs can have the same OSPF area ID if they are in separate VRF instances and are therefore in separate routing domains.

**Note**

Before Cisco APIC, release 2.3(1f), transit routing was not supported within a single L3Out profile. In Cisco APIC, release 2.3(1f) and later, you can configure transit routing within a single L3Out profile, with limitations; for details, see About Transit Routing, on page 49.
Each external router is connected to the same area and learns the same routing information. There is only one L3Out, so route control policies are not needed and there are no issues from a routing perspective. All devices that connect to the Cisco ACI fabric are placed into endpoint groups, including networks reachable through an L3Out. The endpoint group classification for an L3Out is based on configuration policy (it is not based on routing information). In this configuration all peers are configured under the same L3Out and will belong to the same external endpoint group. Even though they are in the same endpoint group traffic will not be permitted unless the prefix classifier is configured for the external endpoint group. This classifier is configured with the External Subnets for the External EPG policy.

The external endpoint group classifier is a longest prefix match classifier. When the subnet 0.0.0.0/0 is configured for the external endpoint group classifier, this will match all traffic between different L3Outs. There is a special case for traffic within the same L3Out. In this case, an implicit deny is configured for traffic between external devices within the same L3Out when using the 0.0.0.0/0 prefix. To allow traffic forwarding through the border leaf switch for traffic within the same L3Out, a more specific prefix classifier must be used.

In the following example, the Cisco ACI border leaf switch will be used for transit traffic between the 192.168.1.0/24 and 172.16.1.0/24 networks.
The 0.0.0.0/0 prefix cannot be used as a classifier due to the default deny rule for this prefix. Therefore, you must create two subnets that will match the external networks.

**EIGRP to EIGRP Transit Routing**

When EIGRP L3Outs are configured on different leaf switches and used for transit routing, mutual redistribution between EIGRP and BGP is performed on the border leaf switches similar to the OSPF use case. One difference is that even though routes are redistributed from EIGRP to BGP, the EIGRP autonomous system information carried in the BGP becomes updated using BGP extended communities. If two EIGRP L3Outs are configured on different nodes but use the same EIGRP autonomous system information, the redistributed routes will appear as local EIGRP routes. This is a key difference in the behavior from OSPF, where all of the redistributed routes appear as OSPF external routes.
The following command output shows the EIGRP to EIGRP transit routing configuration:

```
BL-2# show ip bgp 10.40.1.0/24 vrf hr:ctx1
BGP routing table information for VRF hr:ctx1, address
family IPv4 Unicast
BGP routing table information for 10.40.1.0/24, version 44
Paths: (1 available, best #1)
Flags: (0x80c0002) on xmit-list, is not in urib, exported
vpn: version 550, (0x100002) on xmit-list
Multihop: eBGP iBGP

Advertised path-id 1, VPN AF advertised path-id1
Path type: redist, path is valid, is best path
AS=Path: NONE, path locally originated
    0.0.0.0 (metric 0) from 0.0.0.0 (1.1.1.103)
        Origin incomplete, MED 128576, localpref 100, weight 32768
    Extcommunity:
        RT:65000:2949124
        VNID:2949124
        COST:pre-bestpath:128:128576
        COST:pre-bestpath:162:90
        0x8800:32768:0 (Flags = 32768, Tag = 0)
        0x8801:10:128256 (ASN = 10, Delay = 128256)
        0x8802:65281:320 (Reliability = 255, Hop = 1, Bandwidth = 320)
        0x8803:11:1500 (Reserve = 0, Load = 1, MTU = 1500)
        0x8804:0:0: (Remote ASN = 0, Remote ID = 0)
        0x8805:0:0: (Remote Prot = 0, Remote Metric = 0)
```

The bolded line shows that the EIGRP AS is carried in the BGP extended community.
The route entry on the external router shows the prefix as an internal EIGRP prefix:

```
wan-router# show ip route 10.40.1.0 vrf wan
IP Route Table for VRF "wan"
'*' denotes best unicast next-hop
'**' denotes best mcast next-hop
' [x/y]' denotes [preferences/metric]
'%<string>' in via output denotes VRF <string>

10.40.1.0/24, ubest/mbest: 1/0
   * via 10.1.1.1, eth1/1.22, [90/128832], 00:00:16, eigrp-1, internal, tag 255
```

Transit for BGP L3Outs

The Cisco ACI fabric runs MP-BGP and when routes are propagated across the fabric they will be installed on every leaf (per VRF instance) into the BGP table. When export route control is used to allow transit routes out of a BGP L3Out, an outbound route-map is configured per BGP neighbor. Because export route control for BGP is per neighbor, the BGP L3Outs can be on the same or different border leaf switches. Multiple BGP L3Outs can be on the same border leaf with their own export policies.

Routing Loop Protection with Transit Routing

When L3Outs are configured for transit routing with IGPs (OSPF or EIGRP), mutual route redistribution occurs between BGP and the IGP. Mutual redistribution of routes between different protocols can result in routing loops under certain scenarios. Cisco ACI adds protection for routing loops when redistributing transit routes into EIGRP or OSPF. When a transit route is redistributed into OSPF or EIGRP, the route is tagged with the tag value specified in the route tag policy. If a route is received on an OSPF or EIGRP L3Out with this tag value, the route is dropped. The default route tag policy tag value is 4294967295. The following output shows the tagged route received by the external router, and the table-map and route-map are configured to drop packets with this tag to prevent them from being advertised back into the fabric:

```
172.16.25.0/24, ubest/mbest: 1/0
   * via 192.168.23.0, Eth1/1.23, [110/1], 22:05:25, ospf-1, type-2, tag 4294967295
```

Routing Prcoess default with ID 1.1.1.103 VRF T1:ctx1
Stateful High Availability enabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
Table-map using route-map exp-ctx-3047429-deny-external-tag

```
BL-1# show route-map exp-ctx-2883588-deny-external-tag
route-map exp-ctx-2883588-deny-external-tag, deny, sequence 1
   Match clauses:
      tag: 4294967295
   Set clauses:
      route-map exp-ctx-2883588-deny-external-tag, permit, sequence 2
      Match clauses:
      Set clauses:
```

In some cases, you might not want the routing loop protection. When a transit route from one VRF instance is advertised back into another VRF instance with OSPF or EIGRP, the route will be blocked. The following figure shows that VRF:PN4 is a transit VRF instance and is advertising routes learned from BGP out of OSPF:
These routes will be tagged with tag 4294967295. The L3Out is connected through a firewall back to another L3Out in a different VRF instance. This L3Out also uses the same route tag policy and will block these routes.

The route tag policy can be changed per VRF instance. To change the route tag policy, configure a new route tag policy under protocol policies and assign this policy to the VRF instance.

**Verifying the Transit Routing Configuration**

You can view the transit routing policies in the Application Policy Infrastructure Controller (APIC) GUI. Use following CLI commands on the border leaf switches to verify the routing configuration:

- `show ip ospf vrf vrf_name`
- `show ip eigrp vrf vrf_name`
- `show ip bgp neighbor vrf vrf_name`
- `show route-map`
- `show ip prefix-list`
Additional References for Transit Routing

For more information about Cisco Application Centric Infrastructure (ACI) transit routing support, see the Cisco Application Centric Infrastructure Fundamentals Guide and Cisco APIC and Transit Routing knowledge base article at the following URL:


L3Out Ingress Policy Enforcement

About L3Out Ingress Policy Enforcement

Cisco Application Centric Infrastructure (ACI) uses a whitelist model for security enforcement. A whitelist model requires communication to be explicitly defined before being permitted. The rules used to define what is permitted are configured using contracts, filters, and filter entries. Filter entries specify Layer 4 information.

Contracts are used to permit communications between endpoints that are connected to the ACI fabric. When traffic is sourced from an endpoint, it is identified by a specific group policy ID corresponding to its endpoint group. When one endpoint communicates with another endpoint, the fabric checks whether the group policy ID (source class ID) of the source is permitted to communicate with the group policy ID (destination class ID) of the destination using the specific ports as defined by the filters in the contract.

An endpoint group has a unique class ID. The source class and destination class refer only to relative policy enforcement (which direction is being enforced).

Note

Figure 26: Communication Between Endpoints

Endpoint group classification occurs when a packet arrives on the leaf. For endpoints within the fabric, the classification can be VLAN, VXLAN, MAC address, IP address, VM attribute, and so on. For traffic arriving from an L3Out connection, traffic is classified based on network and mask.
The policy rules (scope, source class ID, dest class ID, and filter) are programmed on the leaf switches in ternary content addressable memory (TCAM).

When a policy is enforced between endpoint groups, it can be enforced on the ingress leaf switch or on the egress leaf switch for internal endpoint groups. On ACI releases prior to 1.2(1), the policy for traffic from an internal endpoint group to an external endpoint group (L3Out endpoint group) is enforced on the egress leaf switch where the L3Out is deployed. A common network design has a large number leaf switches connecting to the compute environment, but only a pair of border leaf switches. Because internal to external policy enforcement is done on the egress switch (border leaf), this can create a resource (TCAM) bottleneck on the border leaf switch.

*Figure 27: Fabric Policy Application Before Release 1.2(1) for Endpoint Group-to-Outside Mapping*

The ingress policy enforcement feature is a configurable option to enable ingress policy enforcement for internal to external communications. With ingress policy enforcement, the destination class lookup for the destination prefix can be done on the ingress leaf switch. This distributes the enforcement of the policy across more switches since there are typically more compute leaf switches than border leaf switches, reducing the likelihood of a bottleneck at the border leaf switches.
Prerequisites for L3Out Ingress Policy Enforcement

You must be using Cisco Application Centric Infrastructure (ACI) release 1.2(1) or later.

Guidelines and Limitations for L3Out Ingress Policy Enforcement

Ingress policy enforcement does not apply to the following cases:

- Transit routing; the rules are already applied at ingress for transit routing
- When a vzAny contract is used
- When a taboo contract is used

Recommended Configuration Procedure for L3Out Ingress Policy Enforcement

Use the ingress policy enforcement when there are a large number of prefixes and external endpoint groups configured at the border leaf switches. Ingress policy enforcement is implemented at the VRF level and applies to all L3Outs that are configured within that VRF.

This feature was introduced in release 1.2(1) and is the default setting for VRFs created in the 1.2(1) release and later. Any VRFs created prior to the release 1.2(1) are set to egress policy enforcement by default and must be manually changed to use ingress policy enforcement.
The following procedure creates a VRF that uses ingress policy enforcement:

**Procedure**

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

**Step 2** In the **Work** pane, double-click the tenant's name.

**Step 3** In the **Navigation** pane, choose **Tenant tenant_name > Networking > VRFs**.

**Step 4** In the **Work** pane, choose **Actions > Create VRF**.

**Step 5** In the **Create VRF** dialog box, on the **VRF** screen, fill in the fields as required, except as specified below:

a) For the **Policy Control Enforcement Direction** buttons, click **Ingress**.

**Step 6** Click **Next**.

**Step 7** On the **Bridge Domain** screen, fill in the fields as required.

**Step 8** Click **Finish**.

---

**Additional References for L3Out Ingress Policy Enforcement**

For more information about ingress policy enforcement, see the *Cisco Application Policy Infrastructure Controller Data Center Policy Model* whitepaper at the following URL:


**L3Out MTU Considerations**

**About L3Out MTU Considerations**

When peering between a Cisco Application Centric Infrastructure (ACI) border leaf switch and an external router, always match MTU values on both sides of the connection. This is especially important when peering using OSPF. During the OSPF neighbor establishment process, each OSPF neighbor sends database descriptor packets (DBD) during the exchange phase. DBD packets include the MTU value of the sending interface. If the MTU value is mismatched between the peers, the neighbors might not reach the **Full adjacency** state.

**OSPF Neighbors Stuck in the Exstart or Exchange State**

A common problem when MTU values are mismatched between OSPF neighbors is that the OSPF adjacency gets stuck in the **Exstart** or **Exchange** state. The following example output is from a Cisco Application Centric Infrastructure (ACI) border leaf for an OSPF adjacency where the MTU values do not match:

```
BL-1# show ip ospf neighbors vrf hr:ctx1
OSPF Process ID default vrf hr:ctx1
Total number of neighbors: 1
Neighbor ID Pri State Up Time Address Interface
1.1.180.1 1 EXSTAT - 00:00:05 20.1.2.0 Eth1/1.57
```
MTU mismatches do not prevent BGP or EIGRP adjacencies from being established, but you should still match MTU values for these peering adjacencies.

**Recommended Configuration Procedure for Setting MTU**

When L3Out interfaces are configured in Cisco Application Centric Infrastructure (ACI), each interface has an MTU setting. The default value for this setting is `inherit`. With this setting, ACI inherits the MTU value that is configured for the fabric Layer 2 MTU policy, which is set to 9000 bytes. You should not change the fabric Layer 2 MTU policy, as this affects all Layer 3 interfaces of the fabric, including all bridge domain subnets. If the MTU for the L3Out interface should be a value other than 9000 bytes, you should change this on the L3Out interface policy.

**Procedure**

1. **Step 1**
   On the menu bar, choose *Tenants > All Tenants*.
2. **Step 2**
   In the *Work* pane, double-click the tenant's name.
3. **Step 3**
   In the *Navigation* pane, choose *Tenant tenant_name > Networking > External Routed Networks*.
4. **Step 4**
   In the *Work* pane, choose *Actions > Create Routed Outside*.
5. **Step 5**
   In the *Create Routed Outside* dialog box, fill in the fields as required, except as specified below:
   - a) On the *Nodes And Interfaces Protocol Profiles* table, click +.
6. **Step 6**
   In the *Create Node Profile* dialog box, fill in the fields as required, except as specified below:
   - a) On the *Interface Profiles* table, click +.
7. **Step 7**
   In the *Create Interface Profile* dialog box, fill in the fields as required, except as specified below:
   - a) In the *Interfaces* section, click the *Routed Sub-Interfaces* button.
   - b) On the *Routed Sub-Interfaces* table, click +.
8. **Step 8**
   In the *Select Routed Sub-Interface* dialog box, fill in the fields as required, except as specified below:
   - a) For the *MTU (bytes)* field, enter the desired MTU value.
   - b) Click *OK*.
9. **Step 9**
   In the *Create Interface Profile* dialog box, click *OK*.
10. **Step 10**
    In the *Create Node Profile* dialog box, click *OK*.
11. **Step 11**
    In the *Create Routed Outside* dialog box, click *Next*.
12. **Step 12**
    Fill in the fields as required and click *Finish*.

**Setting OSPF MTU Ignore**

In some cases, it might not be possible to match the MTU values on both sides of the OSPF peering connection. In this case, you can disable the MTU check when establishing the OSPF adjacency. When there is an MTU mismatch, the side of the connection with the lower MTU value rejects the database descriptor packets (DBD) packets from the neighbor with the higher MTU value because it cannot accept the packet without fragmentation. The MTU ignore setting should be used on the OSPF device with the lower MTU value.

Cisco Application Centric Infrastructure (ACI) also supports the MTU ignore setting using the OSPF interface profile. Use this configuration option if the neighboring OSPF device uses a higher MTU value than the ACI
border leaf switch. If the ACI border leaf switch is sending the higher MTU value, then the MTU ignore setting should be configured on the remote device.

The MTU ignore feature can be used to establish OSPF peer adjacencies when MTU values are mismatched and cannot be modified. This does not affect Path MTU discovery behavior or traffic passing through the border leaf switch. This traffic can still experience fragmentation due to an MTU mismatch. You should match MTU values and only use MTU Ignore in cases where matching is not possible.

The following procedure enables the **MTU Ignore** setting.

**Procedure**

---

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

**Step 2**  
In the **Work** pane, double-click the tenant’s name.

**Step 3**  
In the **Navigation** pane, choose **Tenant tenant_name > Networking > Protocol Policies > OSPF**.

**Step 4**  
In the **Work** pane, choose **Actions > Create OSPF Interface Policy**.

**Step 5**  
In the **Create OSPF Interface Policy** dialog box, fill in the fields as required, except as specified below:  

a) For the **MTU Ignore** check box, put a check in the box.

**Step 6**  
Click **Submit**.

---

**Shared L3Outs**

**About Shared L3Outs**

Using a shared L3Out is an option for a multitenant configuration where each tenant is isolated from each other, but might require access to external shared services, such as DHCP, DNS, and syslog. The Cisco Application Centric Infrastructure (ACI) fabric is very flexible and provides the following options for configuring access to external shared services (shared L3Outs):

1. Create a VRF, bridge domains, and L3Out in the common tenant. Create endpoint groups in individual tenant spaces. In this configuration, tenants share the same VRF and cannot have overlapping IP addresses. All objects created under the common tenant are also visible to each tenant.
2. Create a VRF and L3Out in the common tenant. Create bridge domains and endpoint groups in individual tenant spaces. In this configuration, tenants share the same VRF and cannot have overlapping IP addresses. The bridge domain is configured under the individual tenant spaces and is not visible to other tenants.

3. Create separate tenants with separate VRF instances, bridge domains, and endpoint group. Each tenant has its own VRF instance and can use overlapping IP addresses, as long the overlapping subnets are not leaked into the common tenant. A contract is exported from the tenant that is providing the shared service. Route leaking between VRF instances is performed to provide connectivity between the consumer and provider.
**Prerequisites for Shared L3Outs**

To configure shared L3Outs, you must meet the following prerequisites:

- Use Cisco Application Centric Infrastructure (ACI) release 1.2 or later if the VRF, bridge domain, and subnet are under a user tenant
- Configure a BGP router reflector policy

**Guidelines and Limitations for Shared L3Outs**

The following guidelines and limitations apply for shared L3Outs:

- Transit routing between shared L3Outs in different tenants is not supported.
- Only non-overlapping IP addresses can be leaked between tenants. IP addresses that are not leaked between tenants can overlap.

**Use Cases for Shared L3Outs**

One use case is to have the shared service be external to the fabric and to access the shared service through an L3Out in a tenant (the common tenant or a user tenant):
Other tenants can access this shared service.

In another use case, the shared service is internal and external users access the shared service.

Configuration Example for Shared L3Outs Using the GUI

The following procedure configures a shared tenant to use a shared L3Out:

**Procedure**

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

**Step 2** In the **Work** pane, double-click the shared tenant’s name.

**Step 3** In the **Navigation** pane, choose **Tenant tenant_name > Security Policies > Contracts**.

**Step 4** In the **Work** pane, choose **Actions > Create Contract**.

**Step 5** In the **Create Contract** dialog box, fill in the fields as required, except as specified below:

a) For the **Scope** drop-down list, choose **Global**.

**Step 6** Click **Submit**.

**Step 7** In the **Navigation** pane, choose **Tenant tenant_name > Application Profiles**.

**Step 8** In the **Work** pane, choose **Actions > Create Application Profile**.

**Step 9** In the **Create Application Profile** dialog box, fill in the fields as required, except as specified below:

a) On the **EPGs** table, click + and fill in the fields as required to create an endpoint group.

b) Click **Update**.

**Step 10** Click **Submit**.

**Step 11** In the **Navigation** pane, choose **Tenant tenant_name > Application Profiles > application_profile_name > Application EPGs > application_EPG_name**.

Choose the application profile and application endpoint group that you just created.

**Step 12** In the **Work** pane, choose **Actions > Add Consumed Contract Interface**.
Step 13 In the Add Consumed Contract Interface dialog box, fill in the fields as required, except as specified below:
   a) For the Contract Interface drop-down list, choose the contract interface to export to the consumer tenant.

Step 14 Click Submit.

The L3Out provides the contract and the consumer tenant consumes the contract interface.

Step 15 In the Navigation pane, choose Tenant tenant_name > Application Profiles > application_profile_name > Application EPGs > application_EPG_name > Contracts.

Choose the application profile and application endpoint group that you created in this procedure.

Step 16 In the Work pane, choose Actions > Add Provided Contract.

Step 17 In the Add Provided Contract dialog box, fill in the fields as required, except as specified below:
   a) For the Contract drop-down list, choose the contract that you created in this procedure.

Step 18 Click Submit.

In the Work pane, you can see that the consumer is using the contract interface.

Step 19 In the Navigation pane, choose Tenant tenant_name > Application Profiles > application_profile_name > Application EPGs > application_EPG_name > Subnets.

Choose the application profile and application endpoint group that you created in this procedure.

Step 20 In the Work pane, choose Actions > Create EPG Subnet.

Step 21 In the Create EPG Subnet dialog box, fill in the fields as required, except as specified below:
   a) For the Private to VRF check box, remove the check.
      You do not want to advertise the subnet to the L3Out in its own VRF instance.
   b) For the Advertised Externally check box, add a check.
      You want to advertise the subnet to the L3Out outside of its own VRF instance.
   c) For the Shared between VRFs check box, add a check.
      You want to leak the subnet to the VRF instance in which the provider endpoint group resides.

Step 22 Click Submit.

---

L3Out Router IDs

About L3Out Router IDs

When configuring an L3Out policy in Cisco Application Centric Infrastructure (ACI) for external connectivity, there are a number of managed objects that are created as part of the L3Out configuration. The following diagram shows the L3Out managed objects:
The Logical Node Profile managed object is used to identify the nodes (leaf switches) where the L3Out will be instantiated. The Node managed object is where the node and router ID is configured.

Dynamic routing protocols (OSPF, EIGRP, and BGP) all use the same decision process when assigning a router ID:

1. Manually configure the router ID under the protocol configuration (OSPF, EIGRP, or BGP).
2. If no router ID is configured, use the highest up loopback interface IP address.
3. If no loopback interfaces are configured, use the highest up physical interface IP address.

In ACI, the router ID that is specified in the node profile is always configured as a manual router ID under the protocol that is configured for the L3Out. Therefore, the first selection for the router ID selection process is always used.

**Best Practices for Assigning L3Out Router IDs**

The following best practices apply when assigning L3Out router IDs:
• You should not create 2 separate objects, such as a router ID and a loopback interface, with the same IP address.

The node profile also has an option to create a loopback interface with the same value as the router ID. This option is only needed for BGP if you are establishing BGP peering sessions from a loopback interface with the router ID value. For OSPF and EIGRP, you should disable this option.

---

Note: If the L3Out will be used for Layer 3 multicast (PIM enabled), then always put a check in the Use Router ID as Loopback Address check box.

---

• Create a loopback interface for BGP multi-hop peering between loopback addresses.

For BGP, this option can be enabled if you are peering to the loopback address (BGP multi-hop) and are using the router ID address for the peering. You are not required to peer to the router ID address. You can also establish BGP peers to a loopback address that is not the router ID. For this configuration, disable the Use Router ID as Loopback Address option and specify a loopback address that is different than the router ID.

• Each node (leaf switch) should use a unique router ID.

Do not use the same router ID on different nodes in a single routing domain. Duplicate router IDs can cause routing issues. When configuring L3Outs on multiple border leaf switches, each switch (node profile) should have a unique router ID.

• You should use per-VRF instance router IDs.

• Use the same router ID value for all L3Outs on the same node within the same VRF instance.

When configuring multiple L3Outs on the same node and the same VRF instance, you must use the same router ID value on all L3Outs. Using different router IDs is not supported. A fault will be raised if different router IDs are configured for L3Outs on the same node. If you have multiple VRF instances, you can have per-VRF instance router IDs on the same node.

• Configure a router ID for static L3Outs.

The router ID is a mandatory field for the node policy. It must be specified even if no dynamic routing protocol is used for the L3Out. When creating an L3Out for a static route, you must still specify a router ID value. The Use Router ID as Loopback Address check box should be unchecked and the same rules apply regarding router ID value: use the same router ID for all L3Outs on the same node in the same VRF instance and different router ID for different nodes in the same VRF instance.

The router ID values should be unique in a routing domain. ACI supports separate Layer 3 domains (VRF instances). The router ID should be unique for each node in a VRF instance. The same router ID value can be used on the same node in different VRF instances. If the VRF instances are joined to the same routing domain by an external device then same router ID should not be used in the different VRF instances. The following example shows the two VRF instances joined to the same Layer 3 domain through an external firewall. In this case the router IDs should be different in each VRF instance.
Guidelines and Limitations for L3Out Router IDs

The following guidelines and limitations apply for L3Out router IDs:

- Use the same router ID for all L3Outs on the same node within the same VRF.
- Use a different router ID for each node in the same VRF.
- The router ID value must be a valid IPv4 address in the range of 1.0.0.0 to 223.255.255.255.

Note

The router ID for OSPF and EIGRP is a 32-bit number represented in the IP address format. Both OSPF and EIGRP support router ID values that are not valid IPv4 addresses, such as 0.0.0.1. The router ID for BGP must be a valid IPv4 address. ACI only supports valid IPv4 unicast addresses for router IDs regardless of the protocol used.

Configuration Example for Setting an L3Out Router ID Using the GUI

The following procedure provides an example of configuring an L3Out router ID using the Application Policy Infrastructure Controller (APIC) GUI.

Procedure

Step 1
On the menu bar, choose Tenants > All Tenants.
Step 2
In the Work pane, double-click the tenant's name.

Step 3
In the Navigation pane, choose Tenant tenant_name > Networking > External Routed Networks.

Step 4
In the Work pane, choose Actions > Create Routed Outside.

Step 5
In the Create Routed Outside dialog box, fill in the fields as required, except as specified below:
   a) On the Nodes and Interfaces Protocol Profiles table, click +.

Step 6
In the Create Node Profile dialog box, fill in the fields as required, except as specified below:
   a) On the Nodes table, click +.

Step 7
In the Select Node dialog box, fill in the fields as required, except as specified below:
   a) In the Router ID field, enter a valid IPv4 address in the range of 1.0.0.0 to 223.255.255.255.

Step 8
Click OK.

Step 9
In the Create Node Profile dialog box, click OK.

Step 10
In the Create Routed Outside dialog box, click Next.

Step 11
On the External EPG Networks screen, fill in the fields as required.

Step 12
Click Finish.

Multiple External Connectivity

About Multiple External Connectivity
The ACI fabric provides layer-3 connections to outside networks using L3Out constructs in the ACI fabric. An ACI tenant VRF can have multiple L3Out connections within a single tenant. From a routing perspective, the ACI fabric does not function as a single logical router but rather as a network of routers connected to an MP-BGP core. External networks connect to leaf switches using static routing or dynamic routing protocols and can connect at multiple points to the fabric. These connections can be on the same leaf switch or on different leaf switches.

ACI supports multiple connections to external networks from border leafs. Multiple connections can be made from the same L3Out or from different L3Outs. The decision of when to use the same L3Out or different L3Outs depends on the type of connection. The L3Out managed object is the top-level object for the L3Out and is the container for L3Out logical node profiles and interface profiles. If an L3Out is used to connect to multiple peers on the same or different node they can be configured under the same L3Out or under different L3Outs.

Prerequisites for Multiple External Connectivity
Implementation of multiple external connectivity on the ACI fabric requires that a BGP route reflector policy has been configured.

Guidelines and Limitations for Multiple External Connectivity
ACI supports multiple connections to external networks from border leaf switches. Multiple connections can be made from the same L3Out or from different L3Outs. The decision of when to use the same L3Out or
different L3Outs depend on the type of connection. The L3Out managed object is the top-level object for
the L3Out and is the container for L3Out logical node profiles and interface profiles.

**General Guidelines for Multiple External Connectivity through Multiple or Single L3Out Objects**

- The L3Out object defines the protocol and some protocol parameters that will be used by all nodes and
  interfaces configured under the L3Out.
  - For OSPF L3Outs, the OSPF area is defined at the L3Out level. If an OSPF L3Out will connect to
    multiple external devices on the same border leaf, one L3Out should be configured.
  - Similarly, the EIGRP AS is configured at the L3Out level. If connecting to multiple EIGRP devices
    in the same AS from the same leaf, one L3Out should be used.
  - A different L3Out must be used when connecting to OSPF neighbors in different areas or when
    connecting to EIGRP neighbors in different AS.
  - For BGP L3Outs the peer-connectivity profile is configured under the node (for peering to loopback
    addresses) or under the physical interface (for direct connection peering). Multiple BGP peers can
    be defined under the same L3Out.

- Another decision for single vs multiple L3Outs depends on the type of physical interface.
  - If connecting to multiple external devices on the same VLAN (same subnet) this connection would
    use an L3Out with SVI interfaces.
  - This connection will typically span multiple leaf switches for redundancy.
  - These connections can be on physical ports, port-channels, or virtual port-channels (vPCs).
  - When an L3Out is configured with an SVI, this will create an external bridge domain (VXLAN
    VNI) that is extended across the different switches where the L3Out is deployed.
  - The VLAN/external bridge domain must be configured on a single L3Out. Different L3Outs cannot
    use the same SVI VLAN/external bridge domain.

- When connecting L3Outs to routed or routed sub-interface links, the choice of whether to use one L3Out
  or multiple L3Outs depends on the protocol and security policy requirement.

**Guidelines for Multiple OSPF L3Outs on the same leaf**

- When configuring multiple L3Outs on the same border leaf each L3Out should be in different OSPF
  areas. One area should be area 0 of the border if the leaf will forward transit traffic between the different
  OSPF L3Outs. This follows the OSPF area border router rules that one area must connect to area 0.
- If the same border leaf is connected to multiple OSPF peers in the same area you cannot create separate
  L3Outs. Only one L3Out on a border leaf can be configured in the same area.
- You can configure multiple OSPF peers from a single L3Out. The following example shows a single
  OSPF L3Out configured with multiple OSPF interfaces, which would connect to different peers.

BGP L3Outs are supported on the following connections:

- iBGP over static route
- iBGP over OSPF
• iBGP over direct connection
• eBGP over OSPF
• eBGP over direct connection

When BGP is transported over OSPF for BGP multi-hop connections, the OSPF process that is created on the leaf is only used to learn route the remote BGP peer. OSPF routes in this case are not redistributed into MP-BGP.

BGP over OSPF and regular OSPF L3Outs are not supported on the same leaf.

Guidelines for ensuring Multiple L3Out security

All connections to a Cisco ACI fabric are classified into endpoint groups. The classification for external L3 connections is network/mask and is configured under the external network instance profile, which is also referred to as external EPG. The external EPG is configured under the L3Out and is used for classification or the external EPG, route control, and contract association. Even though the external EPG classification is configured under an L3Out, the classification is applied at the VRF instance level. This should be considered when using multiple L3Outs and overlapping classification rules.

For example, if there are two L3Outs on the same VRF instance and both L3Outs use the 0.0.0.0/0 classification, then traffic coming in from each L3Out can be classified in the same EPG. If an L3Out has an EPG with the overlapping classification but does not have a contract, the traffic may still be permitted by the contract in the other EPG with the same classifier. The example below shows two external EPGs both using 0.0.0.0/0 as the classifier. Traffic coming from L3out-2 is destined to the web EPG but there is no contract. This traffic is still permitted because the classifier 0.0.0.0/0 is configured for the external EPG associated with L3out-1 which does have a contract.
If traffic from L3Out-2 should be blocked from accessing the web EPG, best practice is to use non-overlapping prefixes for the external EPGs and only add classification for the networks that should be permitted to access that service.
Recommended Configuration Procedure for Multiple External Connectivity

To configure an L3Out object with multiple external connectivity, complete the following steps:

**Procedure**

1. **Step 1**
   On the menu bar, choose **Tenants > All Tenants**.
2. **Step 2**
   In the Work pane, double-click the tenant's name.
3. **Step 3**
   In the Navigation pane, choose **tenant_name > Networking > External Routed Networks**.
4. **Step 4**
   In the Work pane, choose **Actions > Create Routed Outside**.
5. **Step 5**
   In the Create Routed Outside dialog box, specify an L3Out name, locate the Nodes and Interface Protocol Profiles table and click + to display the Create Node Profile dialog box.
6. **Step 6**
   In the Create Node Profile dialog box, specify a node profile name, locate the Interface Profiles table, and click + to display the Create Interface Profile dialog box.
7. **Step 7**
   In the Create Interface Profile dialog box, specify an interface profile name and locate the Routed interfaces tab on the Interfaces table. There you can associate the interface profile with multiple routed interfaces. Specify each interface as follows:
   a) Select the Routed Interfaces tab on the Interfaces table and click + to display the Select Routed Interface dialog box.
b) In the Path field, click the drop-down arrow to specify the node and interface to add to the interface profile.
c) In the IPv4 Primary/IPv6 Preferred field, enter the IP address and subnet mask assigned to the interface.
d) Specify any other settings that apply, then click OK.
e) To specify additional routed interface entries, repeat steps a through d.

Step 8  After you finish adding routed interface entries, complete all appropriate fields in the Create Interface Profile dialog box and click OK to save the interface profile.

Step 9  After you save the interface profile, complete all appropriate fields in the Create Node Profile dialog box and click OK to save the node profile.

Step 10  After you save the node profile, complete all appropriate fields in the Create Routed Outside dialog box and click OK to save the L3Out object.

The resulting L3Out object supports external connectivity through multiple interfaces as specified through node profile and interface profile association.
Microsegmentation

About Microsegmentation

Cisco Application Centric Infrastructure (ACI) architecture was designed with multitenancy in mind. ACI has built-in segmentation (with the help of endpoint groups and contracts) and security as part of the architecture, but customers want the ability to secure and segment their data centers and the physical and virtual workloads for more control and manageability reasons. For more granular and dynamic segmentation and to enhance security inside of the data center, the ACI release 1.1(1) added support for microsegmentation.

Interface and VLAN/VXLAN IDs are used for endpoint group classification. In addition, you can use more granular endpoint group derivation based on MAC, IP, or VM information. Even if endpoints are connected to the fabric with a VLAN/VXLAN ID on the same port, you can provide a different security policy for each one. This section describes these microsegmentation capabilities (intra-endpoint group isolation, IP-based endpoint group, and uSeg endpoint group) and how to configure them.

Guidelines and Limitations for Microsegmentation

Application Policy Infrastructure Controller (APIC) supports IP-based endpoint group, uSeg endpoint group, and intra-endpoint group isolation. APIC supports multi-hypervisor virtual endpoints and bare metal endpoints.

Table 1: Endpoint Group Isolation Support

<table>
<thead>
<tr>
<th>Supported APIC Releases</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>uSeg endpoint groups (IP, MAC, VM attribute) for an AVS domain</td>
<td>1.1(1x) or later</td>
</tr>
<tr>
<td>uSeg endpoint groups (IP, MAC, VM attribute) for an SCVMM domain</td>
<td>1.2(1x) or later</td>
</tr>
<tr>
<td>uSeg endpoint groups (IP, MAC, VM attribute) for a VMware vDS domain</td>
<td>1.3(1x) or later</td>
</tr>
</tbody>
</table>
Intra-Endpoint Group Isolation

By default, all endpoints in the same endpoint group can talk to each other without requiring a contract. Intra-endpoint group (intra-EPG) isolation prevents all endpoints in the endpoint group from talking to each other. This is a private VLAN-equivalent feature in a traditional network. Intra-EPG isolation reduces the number of endpoint group encapsulations that you must have when many clients access a common service, but the clients are not allowed to communicate with each other.

Note

Only use this feature when the VRF is in enforced mode, because the feature relies on the correct isolation based on the deployment of contracts.

For example, assume that you have three endpoints: two are in the client endpoint group, while the other endpoint is in the Web endpoint group. If there is a contract between endpoint groups, they can talk each other, as shown in the following figure:

<table>
<thead>
<tr>
<th>Supported APIC Releases</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP-based endpoint groups for a physical domain</td>
<td>1.2(1x) or later</td>
</tr>
<tr>
<td>Intra-endpoint group isolation for VMware vDS and physical domain</td>
<td>1.2(2x) or later</td>
</tr>
<tr>
<td>Intra-endpoint group isolation for AVS domain</td>
<td>1.3(1x) or later</td>
</tr>
</tbody>
</table>
If you enable intra-EPG isolation on the client endpoint group, the endpoints in the endpoint group cannot talk each other, but inter-EPG communication is still permitted if there is a contract, as shown in the following figure:

**Figure 36: Intra-EPG Isolation with a Contract**

<table>
<thead>
<tr>
<th>Callout</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Endpoints in the same endpoint group cannot communicate with one another.</td>
</tr>
<tr>
<td>2</td>
<td>Inter-EPG communication is still permitted if there is a contract.</td>
</tr>
</tbody>
</table>
The backend uses PVLAN (private VLAN). After enabling intra-EPG isolation on the endpoint group, the APIC changes the vDS and port group configuration, and pushes the policy to the physical leaf, which prevents communication between endpoint in the same endpoint group. The following screenshots show this configuration:

By default, you do not need to specify a VLAN encapsulation ID for port groups. The APIC chooses a VLAN from the dynamic VLAN pool that is associated with the VMM domain.

When you use PVLAN, if you have intermediate switches, such as UCS fabric interconnect, between the server and ACI leaf switch, you must configure PVLAN on the intermediate switches. That means that you must confirm which VLAN ID will be used. If you add a static VLAN pool in the VMM domain, you can specify the VLAN ID from the static VLAN pool.

**uSeg Endpoint Group for a Physical Domain**

If you have two endpoints that are in the same VLAN on the same interface and use a VLAN ID and interface for endpoint group classification, the endpoints will be in the same endpoint group. This implies that the endpoints have the same security policy.

*Figure 37: Endpoints with the Same Security Policy*

In the figure, both Server-A and Server-B can connect to both Storage-A and Storage-B.

With an IP-based endpoint group, you can use an IP address for endpoint group classification. For example, 192.168.1.1 is in endpoint group Storage-A and 192.168.1.2 is in endpoint group Storage-B even if they are in the same VLAN and interface. The different endpoint groups enable you to apply different security policies to each endpoint.
Figure 38: Endpoints with Different Security Policies

In the figure, Server-A can only connect to Storage-A, while Server-B can only connect to Storage-B.

To create this configuration, you must create a base endpoint group "Storage" and associate it with a physical domain with static bindings (path or leaf switches). Thus, both 192.168.1.1 and 192.168.1.2 are in the base endpoint group.

Next, create the uSeg endpoint groups "Storage-A" and "Storage-B", which are also associated with a physical domain with static bindings (leaf switches). You can set multiple uSeg attributes in the uSeg endpoint groups. This example uses 192.168.1.1/32 for “Storage-A” and 192.168.1.2/32 for “Storage-B”, but you can specify a larger subnet, such as 172.16.1.0/24.

You must use the following configuration guidelines for the bridge domain and endpoint group setting:

- The base endpoint group and uSeg endpoint group must be in the same bridge domain.
- The bridge domain subnet is required and unicast routing must be enabled because IP-based endpoint group classification applies only for routed traffic.
- Deployment immediacy must be Immediate on the uSeg endpoint group.

uSeg Endpoint Group for a VMM Domain

A uSeg endpoint group for a VMM domain provides the ability to assign virtual endpoints automatically to an endpoint group based on various attributes (MAC address, IP address, and virtual machine information).

If that you have a 3-tier application with several virtual machines in the different endpoint groups and you detect a vulnerability in a particular virtual machine, you can isolate that virtual machine or you can apply a different security policy. Without a uSeg endpoint group, endpoint group classification is based on the port group (VLAN encapsulation ID), and so you must change the virtual machine vNIC to a different port group.

Using a uSeg endpoint group with a virtual machine attribute, you can move the endpoint to the different endpoint group without changing virtual machine vNIC configuration. For example, if the virtual machine name is "Web03," the virtual machine is classified to a uSeg endpoint group, and if the uSeg endpoint group does not have a contract with other endpoint groups, you can isolate the virtual machine. After you determine the cause of the problem, you can delete the attribute configuration on the uSeg endpoint group so that the virtual machine is automatically sent back to the base endpoint group "Web".

The following figure illustrates this scenario:
In the figure, the virtual machine "Web03" is classified in a uSeg EPG, and so the virtual machine "Web03" cannot communicate with other virtual machines.

With the base endpoint group, the uSeg endpoint group can have a contract, and so another use case is migrating the endpoint between different environments. Assume that you are setting up a new application on a server for a test environment and the virtual machine "Test-Webxxx" is in the "Test-Web" endpoint group. Once virtual machine gets ready, you change the virtual machine name to "Prod-Webxxx," which will move the virtual machine to Prod-Web endpoint group.

The following figure illustrates this scenario:

In the figure, the test network and production network are isolated. After changing the virtual machine name, the virtual machine is moved to the production network.

To create this configuration, you must create a base endpoint group and uSeg endpoint group, which are associated with the VMM domain. For example, we have virtual machine "Win7-1" in Base endpoint group "Client" and "Win2012-Web1" in Base endpoint group "Web."

Next, create the uSeg endpoint group "Win2012," which is also associated with the same VMM domain that is specified by the virtual machine attribute. In this example, if virtual machine name contains "2012," it will be in the uSeg endpoint group. Once win2012-Web1 is moved to the uSeg endpoint group, it does not appear
in the base endpoint group "Web." If you remove the uSeg attribute, the virtual machine moves back to the base endpoint group "Web."

You can define multiple types of attributes in the uSeg endpoint group with the following precedences:

**Table 3: uSeg Attribute Precedences**

<table>
<thead>
<tr>
<th>Precedence</th>
<th>Attribute</th>
<th>VMware</th>
<th>Hyper-V</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Mac</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>2</td>
<td>IP</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>3</td>
<td>VNIC (DN)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>4</td>
<td>VM (ID)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>5</td>
<td>VM Name</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>6</td>
<td>Hypervisor</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>7</td>
<td>VMM Domain</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>8</td>
<td>Data center (VMware) Fabric Cloud (Hyper-V)</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>Custom Attribute</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>10</td>
<td>Guest OS</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

When you define string, you can choose one of the following operator types:

- Contains
- Ends With
- Equals
- Starts With

**Additional References for Microsegmentation**

For more information on microsegmentation, see the *Cisco Application Centric Infrastructure Microsegmentation Solution White Paper* document:


See the "ACI Policy Model" chapter of the *Cisco Application Centric Infrastructure Fundamentals Guide*:

Virtualization Design

- VMM Integration with UCS-B, on page 91
- VMM Integration with AVS or VDS, on page 93
- VMM Domain Resolution Immedicacy, on page 96
- OpenStack and Cisco ACI, on page 98

VMM Integration with UCS-B

About VMM Integration with UCS-B

Virtual Machine Manager (VMM) integration allows for the Cisco Application Centric Infrastructure (ACI) fabric to extend network policy and policy group definitions into virtual switches residing on a hypervisor. This integration allows for automation of certain steps that typically create delays in the deployment of virtual and compute resources. The integration is done by allowing the ACI fabric to configure automatically the required fabric side and hypervisor virtual switch encapsulation to ensure matching definitions.

When it comes to ACI and UCS-B interaction, the specific design of the UCS-B has to be taken into account. That is, the fact that the two fabric interconnects are never part of a single logical switch, and the concept of a designated receiver and how the designated receiver is determined on these fabric interconnects. A leaf switch connected to a set of end hosts (compute resources) is commonly referred to as a "leaf node." This terminology will be used throughout this section.

Prerequisites for VMM Integration with UCS-B

Virtual Machine Manager (VMM) integration with UCS-B has the following prerequisites:

- The Virtual Machine Manager (VMM) must be deployed.
- The VMM must be reachable through out-of-band or in-band management from the Application Policy Infrastructure Controller (APIC).
- The VMM must have some hosts integrated into its domain.
- The UCS vNICs must be configured to use either CDP or LLDP. Both protocols cannot be configured, but one is required.
- The block of VLANs to be utilized must be created on UCS and applied only to the leaf node-facing uplinks and the integrated hosts vNICs.
Guidelines and Limitations for VMM Integration with UCS-B

For UCS-B integration, you must take into account the following limitations:

- The only supported OS load balancing mechanism for UCS-B is "Route Based on Originating Virtual Port ID." This equates to the vSwitch policy of "MAC-pinning" within Cisco Application Centric Infrastructure (ACI).

- If utilizing a disjointed Layer 2 domain on the UCS (essentially certain VLANs of certain interfaces), you must have performed proper VLAN pruning on the fabric interconnects. By default, the UCS allows configured VLANs on all interfaces. VLAN trunking is associated with the designated receiver (DR) within the UCS POD. Only one interface (port/port-channel) per VLAN is selected as the DR. There will be endpoint retention issues if the selected DR interface is not one that is connected to the ACI fabric.

- CDP or LLDP is required for most VMM integration deployment scenarios. ACI utilizes these neighbors to resolve virtual host IDs from the end hosts to the leaf nodes. If neighborship is not formed under these scenarios, the leaf node will not push policy to allow for a communication path into the ACI fabric. CDP and LLDP are not required when integrating with Cisco AVS.

Recommended Configuration Procedure for VMM Integration with UCS-B

Although VMM integration aids in configuration by automating the VLAN assignment for both the endpoint group and the port group, there are certain configurations that must still be completed manually or there will be connectivity issues:

**Procedure**

**Step 1**

All intermediate devices should have the dynamic block range of VLANs allowed. In the case of UCS, this means that the user must still navigate the UCS Manager and allow the range of configured VLANs on all VNICs and uplink ports that are going to the ACI fabric.

**Example:**

The design asks to use VLANs 100-200 for VMM integration with UCS-B. The user must go into UCSM and perform the following tasks:

a) Create VLANS 100-200.
b) Allow the VLANs on the Uplink interfaces.
c) Prune the VLANs from undesired uplink interfaces.
d) Allow the VLANs on the vNICs of all hosts that will be integrated.

**Step 2**

In the APIC GUI, create a MAC-pinning port channel policy.

a) On the menu bar, choose **Fabric > Access Policies**.
b) In the **Navigation** pane, choose **Interface Policies > Policies > Port Channel Policies**.
c) In the **Work** pane, choose **Actions > Create Port Channel Policy**.
d) In the **Create Port Channel Policy** dialog box, fill out the fields as necessary

This policy must be associated to the attachable access entity profile as a vSwitch port channel policy to take effect. This only changes the vSwitch port channel policy, not the port channel policy that is associated with the physical interfaces that are utilized by the end hosts.
Step 3  
Associate the port channel policy to the attachable access entity profile as a vSwitch port channel policy.

a) On the menu bar, choose **Fabric > Access Policies**.
b) In the **Navigation** pane, choose **Global Policies > Attachable Access Entity Profiles > AAEP_name**.
c) In the **Work** pane, choose **Actions > Config vSwitch Policies**.
d) In the **Config vSwitch Policies** dialog box, fill out the fields as necessary.

### Verifying the VMM Integration with UCS-B Configuration

**Procedure**

**Step 1**  
Verify the node neighbors by using SSH to connect to the leaf node and run either the **show cdp neighbors** or **show lldp neighbors** command, depending on what configuration is used within this deployment.

**Step 2**  
Verify neighborship directly on the fabric interconnects to ensure that the hypervisor vNICs are forming a neighborship through CDP or LLDP.

**Step 3**  
Verify compute node VLAN programming by using SSH to connect to the node and running the **show VLAN extended** command.

### Additional References for VMM Integration with UCS-B

For additional information on VMM Integration, go to the following URL:


### VMM Integration with AVS or VDS

**About VMM Integration with AVS or VDS**

The integration of Cisco ACI with virtualized servers using a VMware vSphere Distributed Switch (VDS) or Cisco Application Virtual Switch (AVS) provides more control of the virtual environment from the Application Policy Infrastructure Controller (APIC). The APIC aggregates the information from virtualized servers, allowing the administrator to see where virtual machines are located in the fabric, the locations where the virtualized hosts are attached, and more.

With VDS, certain levels of configuration get pushed from the APIC as opposed to manually configuring them directly on the VDS. The configuration pushed from the APIC includes port groups and various port group settings. The VDS on its own can only be deployed utilizing VLANs.

The AVS is derived from the Cisco Nexus 1000v Platform. Similar in control, the APIC pushes port groups, port group settings, and other features that can be utilized, including, but not limited to, the distributed firewall and microsegmentation.
**Prerequisites for VMM Integration with AVS or VDS**

This section lists the prerequisites for Virtual Machine Manager (VMM) integration with AVS or VDS:

- Make a decision on whether or not to use VLAN or VXLAN encapsulation or multicast groups.
- A virtual machine manager must be already deployed, such as vCenter.
- The VMM must be accessible by the Application Policy Infrastructure Controller (APIC) by either out-of-band or in-band management.
- For Cisco Application Virtual Switch (AVS) deployment, a vSphere Installation Bundle (VIB) must be installed on all Hypervisor hosts to be added to the AVS.
- For a VXLAN deployment, know whether or not intermediate devices have Internet Group Management Protocol (IGMP) snooping on or off by default.

**Guidelines and Limitations for VMM Integration with AVS or VDS**

- When utilizing VLANs for VMM integration, regardless of Cisco Application Virtual Switch (AVS) or VMware vSphere Distributed Switch (VDS), the range of VLANs to be used for port groups must be manually allowed on any intermediate devices.
- For VMM integration with VLANs and a resolution immediacy of “On Demand” or “Immediate,” there can be a maximum of one hop in between the hosts and the compute node.
- For VMM integration with VXLAN, only the infrastructure VLAN needs to be allowed on all intermediate devices.
- For VMM integration with VXLAN, if the infra bridge domain subnet is set as a Querier, the intermediate devices must have Internet Group Management Protocol (IGMP) snooping enabled for traffic to pass properly.
  
  Log in to the Advance Mode in the APIC GUI, choose Tenants > Tenant infra > Networking > Bridge Domains > default > Subnets > 10.0.0.30/27

- For VMM Integration with VXLAN and UCS-B, IGMP snooping is enabled on the UCS-B by default. Therefore, ensure that the Querier IP is enabled for the infra bridge domain. The other option is to disable IGMP snooping on UCS and disable the querier IP on the infra bridge domain.

**Verifying the VMM Integration with AVS or VDS**

The following procedures verify that the Cisco Application Virtual Switch (AVS) has been installed on the VMware ESXi hypervisor.

### Verifying the Virtual Switch Status

This section describes how to verify the virtual switch status.

**Procedure**

**Step 1** Log in to the VMware vSphere Client.
Verifying the vNIC Status

This section describes how to verify the vNIC status.

Procedure

Step 1  In VMware vSphere Client, click the Home tab.
Step 2  Choose Hosts and Clusters.
Step 3  Click the host.
Step 4  Click the Configuration tab.
Step 5  In the Hardware panel, choose Networking.
Step 6  In the View field, click the vSphere Distributed Switch button.
Step 7  Click Manage Virtual Adapters. The vmk1 displays as a virtual adapter and lists an IP address.
Step 8  Click the newly created vmk interface to display the vmknic status.

Note  Allow approximately 20 seconds for the vmk to receive an IP address through DHCP.

Additional References for VMM Integration with AVS or VDS

For additional information on virtualization within ACI:


For additional information on ACI Integration and configuration with AVS:


For additional information on ACI integration with VMware:


For additional information on the AVS distributed firewall:

http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/virtualization/b_ACI_Virtualization_Guide_1_3_x/b_ACI_Virtualization_Guide_1_3_x_chapter_0101.html#concept_E89432FC9DDF4F45A3AFB0EA826A7DEA
VMM Domain Resolution Immediacy

About VMM Domain Resolution Immediacy

Resolution immediacy determines at which point to push endpoint group policies to a compute node for end host usage. These policies include VLAN/VXLAN binding, contracts, and filters. Due to the dynamic nature of a VMM domain, most of the policy will wait for an indication of usage (as a trigger) before programming these values. There are certain scenarios where you will want to force programming onto the leaf node before usage. This section discusses both scenarios.

Prerequisites for VMM Domain Resolution Immediacy

The three resolution immediacies are defined as follows:

- **Pre-provision**—Specifies that a policy (for example, VLAN, VXLAN binding, contracts, or filters) is downloaded to a leaf switch even before a hypervisor is attached to the VMware vSphere Distributed Switch (VDS), thereby pre-provisioning the configuration on the switch.

- **Immediate**—Specifies that endpoint group policies (including contracts and filters) are downloaded to the associated leaf switch software upon hypervisor attachment to a VDS. LLDP or OpFlex permissions are used to resolve the hypervisor to leaf node attachments.

- **On Demand**—Specifies that a policy (for example, VLAN, VXLAN bindings, contracts, or filters) is pushed to the leaf node only when a pNIC attaches to the hypervisor connector and a virtual machine is placed in the port group (endpoint group).

Guidelines and Limitations for VMM Domain Resolution Immediacy

At a high level, the least strict definition of a policy comes from the "Pre-Provision" setting. This is essentially a static path, in that the resolution will program the endpoint group policies on all of the interfaces that are linked to that VMM domain as soon as the configuration is made. The resolution is not checking for any level of usage, and will program these interfaces even if the interfaces never get used. This option will pre-provision the VLAN on all ports using the AEP. If an AEP is tied to multiple domains, then the VLAN is pushed to all of the domains in that AEP.

The next level of definition comes from the "Immediate" setting. A resolution set to "Immediate" is only checking for hypervisor attachment to the vSphere Distributed Switch (VDS).

"On-Demand" is the strictest setting, as this has two checks in place to ensure that the policy is only programmed when truly in use. The resolution is looking for the following things:

- Hypervisor attachment to the Application Policy Infrastructure Controller (APIC)-provisioned VDS.
- VM assignment to a port group that was configured from an endpoint group within the APIC.

The value in having a stricter resolution immediacy means that various configurations can be staged from an APIC configuration view without having to worry about resource utilization until truly needed (VM attachment to a port group). However, there are certain virtualization scenarios where this is not ideal and the setting of "Pre-Provision" is truly needed. One such scenario is when migrating a hypervisor management VMK over
to the VDS from a standard vSwitch. Another scenario would be if the NICs of the attached hosts do not support either CDP or LLDP.

**Recommended Configuration Procedure for VMM Domain Resolution Immediacy**

When implementing a VMM domain for virtual machine traffic, a resolution immediacy of “On-Demand” or “Immediate” generally suffices. However, when planning to migrate a hypervisor management VMK over to an in-band VLAN through the VMware vSphere Distributed Switch (VDS), use the “Pre-Provision” immediacy. There are certain configurations that are specific to utilizing “Pre-Provision”:

**Procedure**

**Step 1** Choose a VLAN to be pre-provisioned.

**Step 2** Add the chosen VLAN to a separate range (encap block) within the VLAN pool that is associated with the target VMM domain. The block where this VLAN is added must have the allocation mode set to **Static Allocation**. A static encap block can reside within a dynamic pool specifically for the purpose of using pre-provision.

**Step 3** Create an endpoint group within the desired tenant.

**Step 4** Verify that the bridge domain associated with the management endpoint group is also associated with a VRF.

**Step 5** Associate the VMM domain to the target endpoint group.

**Step 6** Use resolution immediacy **Pre-Provision**.

**Step 7** Specify the management VLAN in the **Port Encap** field of the VM domain profile association.

As a result, the Application Policy Infrastructure Controller (APIC) creates a port group within the VDS with the specified VLAN. The APIC also pushes the endpoint group policies onto the leaf switches that are associated with the VMM domain and Attachable Access Entity Profile (AAEP).

**Verifying the VMM Domain Resolution Immediacy Configuration**

This section describes how to verify the VMM Domain Resolution Immediacy Configuration.

**Procedure**

VLAN programming can be verified by logging into the compute node CLI and running the following command:

```bash
show vlan extended
```

Depending on the immediacy, certain criteria must be met before you will see the VLAN programmed on any interfaces.
OpenStack and Cisco ACI

About OpenStack and Cisco ACI

OpenStack defines a flexible software architecture for creating cloud-computing environments. The reference software-based implementation of OpenStack allows for multiple Layer 2 transports including VLAN, GRE, and VXLAN. The Neutron project within OpenStack can also provide software based Layer 3 forwarding. When utilized with ACI, the ACI fabric provides an integrated Layer 2/3 VXLAN-based overlay networking capability that can offload network encapsulation processing from the compute nodes onto the top-of-rack or ACI leaf switches. This architecture provides the flexibility of software overlay networking in conjunction with the performance and operational benefits of hardware-based networking.

Extending OpFlex to the Compute Node

OpFlex is an open and extensible policy protocol designed to transfer declarative networking policies such as those used in Cisco ACI to other devices. Utilizing OpFlex, the policy model native to ACI can be extended all the way down into the virtual switches running on OpenStack Nova compute hosts. This OpFlex extension to the compute host allows ACI to use Open vSwitch (OVS) to support common OpenStack features such as source NAT (SNAT) and floating IP addresses in a distributed manner.

The ACI OpenStack drivers support two distinct modes of deployment. The first approach is based on the Neutron API and Modular Layer 2 (ML2), which are designed to provide common constructs such as network, router, and security groups that are familiar to Neutron users. The second approach is native to the group-based policy abstractions for OpenStack, which are closely aligned with the declarative policy model used in Cisco ACI.

ACI with OpenStack Physical Architecture

A typical architecture for an ACI fabric with an OpenStack deployment consists of a Nexus 9000 spine/leaf topology, an APIC cluster, and a group of servers to run the various control and compute components of OpenStack. An ACI external routed network connection as a Layer 3 connection outside of the fabric can be used to provide connectivity outside the OpenStack cloud.
OpFlex Software Architecture

The Modular Layer 2 (ML2) framework in OpenStack allows the integration of networking services based on TypeDrivers and MechanismDrivers. Common networking type drivers include local, flat, VLAN, and VXLAN. OpFlex is added as a new network type through ML2, with an actual packet encapsulation of either VXLAN or VLAN on the host defined in the OpFlex configuration. A mechanism driver is enabled to communicate networking requirements from the Neutron servers to the Cisco APIC cluster. The APIC mechanism driver translates Neutron networking elements such as a network (segment), subnet, router, or external network into APIC constructs within the ACI policy model.

The OpFlex software stack also currently utilizes Open vSwitch (OVS), and local software agents on each OpenStack compute host that communicate with the Neutron servers and OVS. An OpFlex proxy from the ACI leaf switch exchanges policy information with the Agent-OVS instance in each compute host, effectively extending the ACI switch fabric and policy model into the virtual switch.
Logical OpenStack Topology

The logical topology diagram in the following figure illustrates the connections to OpenStack network segments from Neutron/controller servers and compute hosts, including the distributed Neutron services.
The management/API network for OpenStack can be connected to servers using an additional virtual
NIC/sub-interface on a common uplink with tenant networking to the ACI fabric, or by way of a separate
physical interface.

Mapping OpenStack and ACI Constructs
Cisco ACI uses a policy model to enable network connectivity between endpoints attached to the fabric.
OpenStack Neutron uses more traditional Layer 2 and Layer 3 networking concepts to define networking
configuration. The OpFlex ML2 driver translates the Neutron networking requirements into the necessary
ACI policy model constructs to achieve the desired connectivity. The **OpenStack GBP Objects and
Corresponding APIC Objects** table illustrates the OpenStack Neutron constructs and the corresponding
APIC policy objects that will be configured when they are created. In the case of GBP deployment, the policies
have a direct mapping to the ACI policy model.

**Table 4: OpenStack Neutron Objects and Corresponding APIC Objects**

<table>
<thead>
<tr>
<th>Neutron Object</th>
<th>APIC Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>(Neutron Instance)</td>
<td>VMM Domain</td>
</tr>
</tbody>
</table>
### Prerequisites for OpenStack and Cisco ACI

This section lists the prerequisites for OpenStack and Cisco ACI:

- **Target audience**—Working knowledge of Linux, intended OpenStack distribution, ACI policy model and GUI-based APIC configuration.
- **ACI fabric**—ACI fabric installed and initialized with a minimum APIC version of 1.1(4e) and NX-OS version of 11.1(4e). For basic guidelines on initializing a new ACI fabric, see the relevant documentation. For communication between multiple leaf pairs, the fabric must have a BGP route reflector enabled to use an OpenStack external network.
- **Servers**—Controller and Compute servers connected to the fabric, preferably using NIC bonding and a vPC. In most cases the Controller does not need to be connected to fabric.
- **L3-Out**—For external connectivity, one or more Layer 3 outs configured on the ACI.
- **VLAN mode**—For VLAN mode, a non-overlapping VLAN pool of sufficient size should be allocated ahead of time.

---

<table>
<thead>
<tr>
<th>Neutron Object</th>
<th>APIC Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project</td>
<td>Tenant + Application Network Profile</td>
</tr>
<tr>
<td>Network</td>
<td>EPG + Bridge Domain</td>
</tr>
<tr>
<td>Subnet</td>
<td>Subnet</td>
</tr>
<tr>
<td>Security Group + Rule</td>
<td>N/A (Iptables rules maintained per host)</td>
</tr>
<tr>
<td>Router</td>
<td>Contract</td>
</tr>
<tr>
<td>Network:external</td>
<td>L3Out/Outside EPG</td>
</tr>
</tbody>
</table>

Table 5: OpenStack GBP Objects and Corresponding APIC Objects

<table>
<thead>
<tr>
<th>GBP Object</th>
<th>APIC Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Target</td>
<td>Endpoint</td>
</tr>
<tr>
<td>Policy Group</td>
<td>Endpoint Group (fvAEPg)</td>
</tr>
<tr>
<td>Policy Classifier</td>
<td>Filter (vzFilter)</td>
</tr>
<tr>
<td>Policy Action</td>
<td>--</td>
</tr>
<tr>
<td>Policy Rule</td>
<td>Subject (vzSubj)</td>
</tr>
<tr>
<td>Policy Ruleset</td>
<td>Contract (vzBrCP)</td>
</tr>
<tr>
<td>L2 Policy</td>
<td>Bridge Domain (fvBD)</td>
</tr>
<tr>
<td>L3 Policy</td>
<td>Context (fvCtx)</td>
</tr>
</tbody>
</table>

---
Guidelines and Limitations for OpenStack and Cisco ACI

This section describes the guidelines and limitations for OpenStack and Cisco Application Centric Infrastructure (ACI).

**Scalability Guidelines**

There is a 1:1 correlation between the OpenStack tenant and the ACI tenant, and for each OpenStack tenant, the plugin automatically creates ACI tenants named according to the following convention:

`convention_apic_system_id_openstack_tenant_name`

You should consider the scalability parameters for supporting the number of required tenants.

Calculate the fabric scale limits for endpoint groups, bridge domains, tenants, and contracts before deployment. Doing so will limit the number of tenant/projects networks and routers that can be created in OpenStack. There are per leaf and per fabric limits. Make sure to check the scalability parameters for the deployed release before deployment. In the case of GBP deployment, it can take twice as many endpoint groups and bridge domains than ML2 mode. The following tables list the Application Policy Infrastructure Controller (APIC) resources that are needed for each OpenStack resource in GBP and ML2 configurations.

**Table 6: OpenStack GBP and ACI Resources**

<table>
<thead>
<tr>
<th>GBP Resource</th>
<th>APIC Resources Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>L3 Policy</td>
<td>1 context</td>
</tr>
<tr>
<td>L2 Policy</td>
<td>1 bridge domain</td>
</tr>
<tr>
<td></td>
<td>1 endpoint group</td>
</tr>
<tr>
<td></td>
<td>2 contract</td>
</tr>
<tr>
<td>Policy Group</td>
<td>1 endpoint group</td>
</tr>
<tr>
<td>Ruleset</td>
<td>1 contract</td>
</tr>
<tr>
<td>Classifier</td>
<td>2 filters (forward and reverse)</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong> 5 overhead classifiers are created.</td>
</tr>
</tbody>
</table>

**Table 7: OpenStack ML2 and ACI Resources**

<table>
<thead>
<tr>
<th>ML2 Resource</th>
<th>APIC Resources Consumed</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network</td>
<td>1 bridge domain</td>
</tr>
<tr>
<td></td>
<td>1 endpoint group</td>
</tr>
<tr>
<td>Router</td>
<td>1 contract</td>
</tr>
<tr>
<td>Security Groups</td>
<td>N/A (no filters are used)</td>
</tr>
</tbody>
</table>
Availability Guidelines

For redundancy, use bonded interfaces (vPCs) by connecting 2 interfaces to two leaf switches and creating a vPC in ACI.

You should deploy redundant OpenStack controller nodes to avoid a single point of failure.

The external network should also be designed to avoid a single point of failure and service interruption.

NAT/External Network Operations

The OpFlex driver software brings the capability to support external network connectivity and Network Address Translation (NAT) functions in a distributed manner using the local OVS instance on each OpenStack compute node. This distributed approach increases the availability of the overall solution and offloads the central processing of NAT from the Neutron server Layer 3 agent that is used in the reference implementation. You can also provide direct external connectivity without NAT or with a mix of NAT and non-NAT external connectivity.

Subnets Required for NAT

Contrary to the standard Neutron approach, three distinct IP subnets are required to take full advantage of external network functionality with the OpFlex driver.

- Link Subnet—This subnet represents the actual physical connection to the external next-hop router outside of the fabric to be assigned to a routed interface, sub-interface, or SVI.
- Source-NAT Subnet—This subnet is used for Port Address Translation (PAT), allowing multiple virtual machines to share an outside-routable IP address. A single IP address is assigned to each compute host and Layer 4 port number manipulation is used to maintain unique session traffic.
- Floating IP Subnet—The term "floating IP" in OpenStack is used when a virtual machine instance is allowed to claim a distinct static NAT address to support inbound connections to the virtual machine from outside of the cloud. The floating IP subnet is the subnet assigned within OpenStack to the Neutron external network entity.

For information about the external connectivity in OpFlex plugin, see the Cisco ACI with OpenStack OpFlex Architectural Overview document:


Optimized DHCP and Metadata Proxy Operations

The OpFlex driver software stack provides optimized traffic flow and distributed processing to provide DHCP and metadata proxy services for virtual machine instances. These services are designed to keep as much processing and packet traffic local to the compute host. The distributed elements communicate with centralized functions to ensure system consistency. You should enable optimized DHCP and metadata services when deploying the OpFlex plugin for OpenStack.

For information about how these optimized services work, see the Cisco ACI with OpenStack OpFlex Architectural Overview document:

Physical Interfaces

OpFlex uses the untagged fabric interface for an uplink trunk in VLAN mode. This means the fabric interface cannot be used for PXE, since PXE usually requires an untagged interface. If you require PXE in a VLAN mode deployment, you must use a separate interface for PXE. This interface can be connected through ACI or an external switch. This issue is not present in VXLAN mode since tunnels are created using the tagged interface for infra VLAN.

Layer 4 to Layer 7 Services

Service insertion in OpenStack is done through a physical domain or device package. Check customer requirements and the plugin mode (GBP, ML2) to plan how service insertion/chaining will be done. The OpenStack Neutron project also defines Layer 4 to Layer 7 extension APIs, such as LBaaS, FWaaS, and VPNaaS. The availability of these extensions depends on device vendors. Check the vendor for the availability of these extensions.

Blade servers

When deploying on the blade servers, you must make sure there is no intermediate switch between the fabric and the physical server interfaces. Check the OpenStack ACI plugin release notes to make sure the configuration is supported. At the time of this writing, there is limited support for B-Series blade servers and the support is limited to VLAN mode only.

Verifying the OpenStack Configuration

The following procedure verifies the OpenStack configuration:

Procedure

Step 1 Verify that a VMM domain was created for the OpenStack system ID defined during installation. The nodes connected to the fabric, running OpFlex agent, should be visible under Hypervisors. The virtual machines running on the hypervisor should be visible upon selecting that hypervisor. All networks created for this tenant should also be visible under the DVS submenu and selecting the network should show you all endpoints connected to that network.

Step 2 Look at the health score and faults for the entity to verify correct operation. If the hypervisors are not visible or show as disconnected, check the OpFlex connectivity.

Step 3 Verify that there is a tenant created for the OpenStack tenant/project. All of the networks created in OpenStack should show up as endpoint groups and corresponding bridge domains. Choose the Operational tab for the endpoint group to show all of the endpoints for that endpoint group.

Step 4 Choose the Health Score tab and Faults tab to make sure that there are no issues.

Configuration Examples for OpenStack and Cisco ACI

This section describes the configuration examples for OpenStack and Cisco Application Centric Infrastructure (ACI).
Optimized Metadata and DHCP

In the configuration file, the optimized DHCP is enabled by default in OpFlex OpenStack plugin. To disable the optimized DHCP, add the following line:

```python
enable_optimized_dhcp = False
```

In the configuration file, the optimized metadata service is disabled by default. To enable the optimized metadata, add the following line:

```python
enable_optimized_metadata = True
```

For more information, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Ubuntu:


For more information, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Red Hat:


External Network/NAT Configuration

External network connectivity is defined by adding "apic_external_network" section to the configuration file. For example:

```yaml
[apic_external_network:DC-Out]
preexisting=True
external_epg=DC-Out-EPG
host_pool_cidr=10.104.11.1/24
```

The `host_pool_cidr` defines the SNAT subnet. The floating IP subnet is defined by creating an external network in Neutron, or an external policy in GBP. The name of the external network or policy should use the same name as "apic_external_network" defined in the file (in this case "DC-Out").

It is possible to disable NAT by adding `enable_net = False` in the above section. You can have multiple external networks using different Layer 3 Out on ACI, and have a mix of NAT and non-NAT external networks.

For more information on external network configuration, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Ubuntu:


For more information on external network configuration, see the Cisco ACI with OpenStack OpFlex Deployment Guide for Red Hat:


Network configuration for GBP

In GBP deployment, network subnets for policy groups are carved out of the `default_ip_pool` defined in the plugin configuration file. For example:

```yaml
[group_policy_implicit_policy]
default_ip_pool = 192.168.0.0/16
```

The above pool will be used to allocate networks for created policy groups. You must make sure that the pool is large enough for the intended number of groups.
Additional references for Openstack and Cisco ACI

For more information, see the following documents:

• Cisco ACI with OpenStack OpFlex Architectural Overview

• Cisco ACI with OpenStack OpFlex Deployment Guide for Ubuntu

• Cisco ACI with OpenStack OpFlex Deployment Guide for Red Hat

• Cisco ACI Installation Guide for Mirantis OpenStack
Additional references for Openstack and Cisco ACI
Layer 4 to Layer 7 Design

- Service Graphs and Layer 4 to Layer 7 Services Integration, on page 109
- Firewall Service Graphs, on page 113
- Service Node Failover, on page 117
- Service Graphs with Multiple Consumers and Providers, on page 119
- Reusing a Single Layer 4 to Layer 7 Device for Multiple Service Graphs, on page 125
- Service Graphs with Route Peering, on page 128
- The Common Tenant and User Tenants, on page 135

Service Graphs and Layer 4 to Layer 7 Services Integration

About Service Graphs and Layer 4 to Layer 7 Services Integration

A Cisco Application Centric Infrastructure (ACI) service graph provides automation for Layer 4 to Layer 7 services deployment in the network. You can deploy Layer 4 to Layer 7 services, such as firewalls and load balancers, with ACI with or without the service graph. To decide whether or not you should use a service graph, you must understand the use case and the operational model that you want to achieve and also the solution that a service graph can provide.

Layer 4 to Layer 7 Services Integration Options

The following Layer 4 to Layer 7 services integration options exist:

- Service graph with managed mode (network and Layer 4 to Layer 7 device configuration automation)
- Service graph with unmanaged mode (network-only stitching)
- Traditional endpoint group stitching by using an endpoint group for service node interfaces (no service graph is required)

For example, you might find the service graph useful if you want to create a portal from which administrators can create and decommission network infrastructure. The portal includes the configuration of firewalls and load balancers. In this case, a service graph with managed mode can automate the configuration of the firewall and load balancers and expose the firewall and load balancers to the portal using the Application Policy Infrastructure Controller (APIC) API. To use a service graph with managed mode, you need a device package for the service node.
With the service graph with managed mode, the configuration of the Layer 4 to Layer 7 device is part of the configuration of the entire network infrastructure. You must consider the security and load balancing rules at the time that you configure network connectivity for the Layer 4 to Layer 7 device. This approach is different from that of traditional service insertion in that if you do not use the service graph, you can configure the security and load balancing rules at a different time than when you configure network connectivity.

If you prefer to manage the configuration of the firewalls and load balancers by using an existing method, such as by using the CLI, GUI, and API of the service device directly, because of the current operation model, a service graph with unmanaged mode is a good option. Since the APIC does not configure the service node itself, a device package is not required for unmanaged mode.
Table 9: Callouts for Service Graph with Unmanaged Mode

<table>
<thead>
<tr>
<th>Callout</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Configure the Cisco Application Centric Infrastructure (ACI) fabric for the Layer 4 to Layer 7 service appliance.</td>
</tr>
<tr>
<td>2</td>
<td>Configure the Layer 4 to Layer 7 service appliance.</td>
</tr>
</tbody>
</table>

If all that you need is a topology with a perimeter firewall that controls the access to the data center from external servers, and if this firewall is not decommissioned and provisioned again periodically, then a service graph is not necessary. You can create endpoint groups for firewall interfaces and configure the contracts so that the client endpoint can access the firewall external interface and the firewall internal interface can access the web endpoint. In this configuration, communication between the client and web occurs through the firewall, as shown in the following figure:

Figure 46: No Service Graph (Using an Endpoint Group as a Service Node)

When to Use a Service Graph for Layer 4 to Layer 7 Services Integration

A service graph offers several advantages and some disadvantages. The advantages are as follows:

- The configuration template can be reused multiple times
- A service graph provides a more logical view and offers an application-related view of services
- You can use a service graph to provision a device that is shared across multiple departments
- A service graph automatically manages VLAN assignments
- A service graph automatically connects virtual network interface cards (vNICs)
- A service graph collects health scores from the device or service
- A service graph collects statistics from the device
- A service graph updates ACLs and pools automatically with endpoint discovery
- You can use the unmanaged mode to avoid using a device package
The disadvantages are as follows:

- The topology is restricted. For example, a service graph is always associated with a contract, which means that the topology always uses a provider-consumer relationship.
- The operational model is orientated toward automation.

When choosing whether to use a service graph or traditional bridge domain stitching, you must take into account the following points:

- Do you need the firewall and load balancers to be configured dynamically through the Application Policy Infrastructure Controller (APIC), or should a different administrator configure them? In the second case, you should not use the service graph with managed mode.
- Do you need to be able to commission, use, and decommission a firewall or a load balancer frequently, as in a cloud service, or will these services be used in the same way for a long period of time? In the second case, you might not see much advantage in using a service graph.

The following flowchart shows how to choose the service graph deployment method:

*Figure 47: Service Graph Decision Flowchart*

---

**Additional References for Layer 4 to Layer 7 Services Integration**

For more information about service graphs, see the *Service Graph Design with Cisco Application Centric Infrastructure White Paper* at the following URL:

Firewall Service Graphs

About Firewall Service Graphs

Service graph deployment can be configured with one of the following modes:

- **GoTo**—The Layer 4 to Layer 7 device is a Layer 3 device that routes traffic; the device can be used as the default gateway for servers or the next hop
- **GoThrough**—The Layer 4 to Layer 7 device is a transparent Layer 2 device; the next-hop or the outside bridge domain provides the default gateway

Prerequisites for a Firewall Service Graph

To configure firewall service graph with managed mode, you must meet the following prerequisites:

- Basic Cisco Application Centric Infrastructure (ACI) setup must be complete. This means you must have created the following objects for a service node:
  - Attachable entity profile
  - Tenant
  - VRF
  - Bridge domain
  - Endpoint group
  - VMM domain or physical domain

- If you are using Cisco ASA, then ASA must be deployed on an ESXi that is participating in a VMware vDS VMM domain.

Recommended Configuration Procedure for a Firewall Service Graph

The following procedure uses an example of a Cisco ASA service graph in the routed mode, which has a firewall between two endpoint groups in the same VRF.
The procedure assumes that the VRF, bridge domains, and endpoint groups are already created.

**Procedure**

**Step 1** On the menu bar, choose L4-L7 Service > Packages.

**Step 2** In the Work pane, click Import a Device Package.

**Step 3** In the Import Device Package dialog, click Browse.

**Step 4** Navigate to the Cisco ASA device package and click Open.

**Step 5** Click Submit.

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

**Step 7** In the Work pane, double-click the tenant's name.

**Step 8** In the Navigation pane, choose Tenant tenant_name > L4-L7 Services > L4-L7 Devices.

**Step 9** In the Work pane, choose Actions > Create L4-L7 Devices.

**Step 10** In the Create L4-L7 Devices dialog, perform the following actions:

- In the General section:
  - In the Name field, enter a name for the device.
  - In the Service Type drop-down list, choose Firewall.
  - For the Device Type buttons, click VIRTUAL.
  - In the VMM Domain drop-down list, choose VMM_Domain.
  - In the Device Package drop-down list, choose the Cisco ASA device package.
  - In the Model drop-down list, choose ASAv.
  - For the Function Type buttons, click GoTo for routed mode.

- In the Connectivity section:
  - For the APIC to Device Management Connectivity radio buttons, click Out-Of-Band. However, if you use in-band management for Application Policy Infrastructure Controller (APIC) service node communication, click In-Band instead.
In the **Device** section:

- In the **Management IP Address** field, enter your ASAv management IP address.
- In the **Management Port** field, enter 443.
- In the **VM Name** drop-down list, choose your ASAv virtual machine.
- In the **Device Interfaces** table, add concrete interfaces for the external and consumer endpoint groups. If you do not use route-peering, you do not need to choose a path.

In the **Cluster** section:

- In the **Management IP Address** field, enter your ASAv management IP address.
- In the **Management Port** field, enter 443.
- In the **VM** drop-down list, choose your ASAv virtual machine.
- In the **Cluster Interfaces** table, add a mapping to the concrete interface that is used for the consumer/provider.

**Step 11** Click Next.

**Step 12** (Optional) In the **Device Configuration** screen, if you need a specific device configuration, such as a failover configuration, define the parameters.

**Step 13** Click Finish.

**Step 14** In the **Navigation** pane, choose **Tenant tenant_name > L4-L7 Services > L4-L7 Devices > device_name**. Choose the device that you just created.

**Step 15** In the **Work** pane, in the **Configuration State** section, ensure that the **Device State** is **Stable** before proceeding with this procedure.

**Step 16** In the **Navigation** pane, choose **Tenant tenant_name > L4-L7 Services > L4-L7 Service Graph Template**.

**Step 17** In the **Work** pane, choose **Actions > Create L4-L7 Service Graph Template**.

**Step 18** In the **Create L4-L7 Service Graph Template** dialog box, perform the following actions:

- In the **Graph Name** field, enter a name for the service graph template.
- Drag and drop the Layer 4 to Layer 7 device that you created from **Device Clusters** section to the graph.
- For the **Firewall** radio buttons, click **Routed** or **Transparent** as appropriate for your desired configuration.
- In the **Profile** drop-down list, choose a function profile.

**Step 19** Click Submit.

**Step 20** In the **Navigation** pane, choose **Tenant tenant_name > L4-L7 Services > L4-L7 Service Graph Template > template_name**.

**Step 21** Right click the service graph template and choose **Apply Service Graph Template**.

**Step 22** In the **Apply Service Graph Template** dialog, perform the following actions:

In the **EPGs Information** section:

- In the **Consumer EPG / External Network** drop-down list, choose the consumer EPG where you want to insert ASAv.
• In the **Provider EPG / External Network** drop-down list, choose the provider EPG where you want to insert ASAv.

In the **Contract Information** section, you can either choose an existing contract where you want to attach the service graph, or you can create a new one.

**Step 23**
Click Next.

**Step 24**
In the **ASAv Parameters** screen, define the Layer 4 to Layer 7 parameters.

**Example:**
As an example, define the following parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device Config &gt; externalIf &gt; externalIfCfg &gt; IPv4Address</td>
<td>192.168.1.101/255.255.255.0</td>
</tr>
<tr>
<td>Device Config &gt; internalIf &gt; internalIfCfg &gt; IPv4Address</td>
<td>192.168.2.101/255.255.255.0</td>
</tr>
</tbody>
</table>

**Step 25**
Click Finish.

The APIC attaches the service graph to the contract and creates the devices selection policies.

---

**Verifying a Firewall Service Graph Using the GUI**

The following procedure verifies that a firewall service graph deployed successfully, using a Cisco ASA two-arm service graph as the example.

**Procedure**

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

**Step 2**
In the **Work** pane, double-click the tenant's name.

**Step 3**
In the **Navigation** pane, choose **Tenant tenant_name > L4-L7 Services > Deployed Devices > device_name**.

**Step 4**
In the **Work** pane, view the properties of the device and check which VLANs are assigned to the service node interface.

**Step 5**
In the vCenter GUI, verify that the port groups were created and that the automatic vNIC placement was performed.

**Step 6**
On the Cisco ASA, verify that the configuration is correct.

**Example:**
```
interface GigabitEthernet0/0
   nameif internalIf
   security-level 100
   ip address 192.168.2.101 255.255.255.0
!
interface GigabitEthernet0/1
   nameif externalIf
```
Additional References for a Firewall Service Graph

For more information about deploying a firewall service graph, see the Cisco APIC Layer 4 to Layer 7 Service Graph Deployment Guide at the following URL:


Service Node Failover

About Service Node Failover

Having a redundancy of service devices improves availability. Each service device vendor has different failover link options and mechanisms. Typical options are as follows:

• Dedicated physical interface for failover traffic, such as F5 devices: the service device has a dedicated physical interface for failover traffic, only.

• Created failover VLAN and interface, such as Cisco ASA devices: the service device does not have a dedicated physical interface. Create a failover VLAN or choose interfaces for failover traffic, which typically are created on different physical interfaces, with one for data traffic.

• Shared (not dedicated) VLAN and logical interface, such as Citrix devices: failover traffic is exchanged over the same VLAN as data traffic.

Typically, use of a dedicated physical interface and a directly cabled pair of failover devices is recommended. If failover interfaces are connected to each service device directly, Cisco Application Centric Infrastructure (ACI) fabric does not need to manage the failover network. If you prefer to have in-band failover traffic within the ACI fabric, create an endpoint group for failover traffic.
If you use a physical appliance and you prefer in-band failover traffic, create an endpoint group for failover using static bindings. This case is similar to the bare metal endpoint case.

If you use a virtual appliance and you prefer to use out-of-band failover traffic, create a port group manually and use it. If you prefer in-band failover traffic, create an endpoint group for failover using a VMM domain, which is similar to the virtual machine endpoint case.
Figure 50: Virtual Appliance with In-Band Failover Traffic

Service Node Failover

For more information about service graph design, see the Service Graph Design with Cisco Application Centric Infrastructure White Paper at the following URL:


Service Graphs with Multiple Consumers and Providers

About Service Graphs with Multiple Consumers and Providers

You can deploy a service graph that has single and multiple consumers and providers. The Cisco Application Centric Infrastructure (ACI) security policy gets updated when you deploy the service graph.
Configuration Example of a Security Policy Before and After Deploying a Service Graph

In the Cisco Application Centric Infrastructure (ACI) fabric, a security policy is applied based on the source class, destination class, and filter matching. The rule is programmed on a leaf.

The procedure in this section assumes that you have the following contract that is between two endpoint groups (EPGs) in the same VRF whose segment scope is 3112960:

Figure 51: VRF Topology Before Applying the Service Graph

The tenant is named "T1".

The procedure shows a security policy before and after you deploy a service graph.

Procedure

**Step 1**
In the advanced GUI, on the menu bar, choose Tenants > All Tenants.

**Step 2**
In the Work pane, double-click T1.

**Step 3**
In the Navigation pane, choose Tenant T1 > Networking > VRFs > VRF1.

**Step 4**
In the Work pane, search for the Segment field to find the VRF segment scope ID. Ensure that the ID is correct.

**Step 5**
In the Navigation pane, choose Tenant T1 > Application Profiles > ANP > Application EPGs > EPG Client.

**Step 6**
In the Work pane, search for the pcTag(sclass) field to find the endpoint group class ID. Ensure that the ID is correct.

**Step 7**
In the Navigation pane, choose Tenant T1 > Application Profiles > ANP > Application EPGs > EPG Web.

**Step 8**
In the Work pane, search for the pcTag(sclass) field to find the endpoint group class ID.

**Step 9**
In the CLI, run the show zoning-rule command. Leafs have a zoning rule that permits the traffic between this source endpoint group and destination endpoint group.

**Example:**

Leaf1# show zoning-rule

<table>
<thead>
<tr>
<th>Rule ID</th>
<th>SrcEPG</th>
<th>DstEPG</th>
<th>FilterID</th>
<th>operSt</th>
<th>Scope</th>
<th>Action</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Step 10

Apply the service graph.

For more information, see the *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide* at the following URL:


Once you have applied the service graph, regardless of managed mode or unmanaged mode, the zoning rule will be updated automatically based on the service graph configuration.

*Figure 52: VRF Topology After Applying the Service Graph*

---

Step 11

To see the updated zoning rules, in the CLI, run the `show system internal policy-mgr stats` command.

**Example:**

```
Leaf1# show system internal policy-mgr stats | grep 3112960
Rule (4104) DN (sys/actrl/scope-3112960/rule-3112960-s-49155-d-16390-f-default)
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4105) DN (sys/actrl/scope-3112960/rule-3112960-s-16390-d-49155-f-default)
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4106) DN (sys/actrl/scope-3112960/rule-3112960-s-32772-d-49154-f-default)
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4107) DN (sys/actrl/scope-3112960/rule-3112960-s-49154-d-32772-f-default)
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
```

The service node is in the middle between consumer and provider endpoint group. If you have only one contract subject to which the service graph is applied, there is no permit rule between the Client endpoint group (49154) and Web endpoint group (49155). In this case, the endpoint groups cannot talk to each other directly.

Step 12

If you want to allow specific traffic between the Client endpoint group and Web endpoint group even after applying a service graph, use two subjects under the contract.
You must use policy-based redirect and the Application Delivery Controller (ADC) with SNAT as a virtual IP address. The real server IP address can be on a different bridge domain and subnet.

As an example, assume that you have Subject1 and Subject2 under the contract with the following configurations:

- Subject1—permit ICMP without a service graph
- Subject2—permit all with a service graph

In this case, the zoning rule allows ICMP traffic between the Client endpoint group (49154) and Web endpoint group (49155).

*Figure 53: ICMP Traffic Between the Client Endpoint Group and Web Endpoint Group*

**Design**

a) To see the zoning rules that allow ICMP traffic between the Client endpoint group (49154) and Web endpoint group (49155), in the CLI, run the `show system internal policy-mgr stats` command.

**Example:**

```
Leaf1# show system internal policy-mgr stats | grep 3112960
...  
Rule (4104) DN (sys/actrl/scope-3112960/rule-3112960-s-49155-d-16390-f-default)
   Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4105) DN (sys/actrl/scope-3112960/rule-3112960-s-16390-d-49155-f-default)
   Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4106) DN (sys/actrl/scope-3112960/rule-3112960-s-32772-d-49154-f-default)
   Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4107) DN (sys/actrl/scope-3112960/rule-3112960-s-49154-d-32772-f-default)
   Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4108) DN (sys/actrl/scope-3112960/rule-3112960-s-49154-d-49155-f-5)
   Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
Rule (4109) DN (sys/actrl/scope-3112960/rule-3112960-s-49155-d-49154-f-5)
   Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
```
Step 13
Before you apply the service graph, if you have multiple consumer and provider endpoint groups for the contract, zoning rules are created for each consumer and provider endpoint group combination.

*Figure 54: Zoning Rules for the Consumers Endpoint Groups and Provider Endpoint Groups*

To see the zoning rules that are created, in the CLI, run the `show system internal policy-mgr stats` command.

**Example:**

```
Leaf1# show system internal policy-mgr stats | grep 3112960
...  
Rule (4122) DN (sys/actrl/scope-3112960/rule-3112960-s-49154-d-49159-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
Rule (4123) DN (sys/actrl/scope-3112960/rule-3112960-s-49159-d-49154-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
Rule (4124) DN (sys/actrl/scope-3112960/rule-3112960-s-49154-d-49155-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
Rule (4125) DN (sys/actrl/scope-3112960/rule-3112960-s-49155-d-49154-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
 Rule (4126) DN (sys/actrl/scope-3112960/rule-3112960-s-49159-d-49158-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
 Rule (4127) DN (sys/actrl/scope-3112960/rule-3112960-s-49158-d-49159-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
 Rule (4128) DN (sys/actrl/scope-3112960/rule-3112960-s-49159-d-49158-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0  
 Rule (4129) DN (sys/actrl/scope-3112960/rule-3112960-s-49158-d-49159-f-default)  
  Ingress: 0, Egress: 0, Pkts: 0  RevPkts: 0
```

Step 14
After applying the service graph with multiple consumers and providers, the service graph updates the rule to insert service nodes between endpoint groups.
**Step 15** Check the class ID for service nodes in the deployed device.

- **a)** On the menu bar, choose *Tenants > All Tenants*.
- **b)** In the *Work* pane, double-click *T1*.
- **c)** In the *Navigation* pane, choose *Tenant T1 > L4-L7 Services > Deployed Devices > ASA-VRF1*.
- **d)** In the *Work* pane, you can see the resource (class) IDs.
Reusing a Single Layer 4 to Layer 7 Device for Multiple Service Graphs

About Reusing a Single Layer 4 to Layer 7 Device for Multiple Service Graphs

A Layer 4 to Layer 7 device defined in the Application Policy Infrastructure Controller (APIC) can be used for multiple service graph instantiations. This section describes how to reuse a single Layer 4 to Layer 7 device for multiple service graphs.

Prerequisites for Reusing a Single Layer 4 to Layer 7 Device for Multiple Service Graphs

You must meet the following prerequisites to reuse a single Layer 4 to Layer 7 device for multiple service graphs:

- The basic Application Policy Infrastructure Controller (APIC) configuration (tenant, VRF, bridge domain, and VMM domain or physical domain for Layer 4 to Layer 7 devices) must be set up.
- For a managed mode service graph, an appropriate device package must be uploaded to the APIC.
- The service graph templates must be set up.
- If the device is not sharing a single physical appliance with multi-contexts, then you can use one physical or virtual appliance as a Layer 4 to Layer 7 device and share the device for multiple service graph instantiations.

Guidelines and Limitations for Reusing a Single Layer 4 to Layer 7 Device for Multiple Service Graphs

You can create multiple cluster interfaces on a concrete device and then specify which cluster interface that is defined in the Layer 4 to Layer 7 device will be used for the connector in the device selection policy. This cluster interface can be shared by using multiple service graph instantiations.

In the Application Policy Infrastructure Controller (APIC) release 2.0 and earlier, port group VLAN trunking for virtual appliance is not supported. If you use a virtual appliance as a Layer 4 to Layer 7 device and you need to add service node interfaces in a different bridge domain, you must have different cluster interfaces on the virtual appliance.

For the endpoint groups, the Layer 4 to Layer 7 device and service graph templates are within one tenant in the following example. The Layer 4 to Layer 7 device that is defined in a tenant cannot be referenced from other tenants. If you want to share a Layer 4 to Layer 7 device with other tenants, export the Layer 4 to Layer 7 device to other tenants. The device will appear as an imported device in the other tenants.
Configuration Example for a Virtual Appliance That is Used By Multiple Service Graphs

The following figure shows a configuration example of a Cisco ASAv virtual device that has three interfaces that are used by two service graphs:

*Figure 56: Cisco ASAv Virtual Device with Three interfaces That are Used by Two Service Graphs*

The following steps provide information about creating a Layer 4 to Layer 7 device with shared interfaces to prepare a virtual appliance to be used by multiple service graphs.

**Procedure**

**Step 1**
Create a Layer 4 to Layer 7 device.

Add the following cluster interfaces:

- External (for subnet 192.168.1.0/24) as consumer
- DMZ (for subnet 192.168.2.0/24) as provider and consumer
- Internal (for subnet 192.168.3.0/24) as provider

The Cisco ASA DMZ interface (192.168.2.1) is the consumer and also the provider, and so you must choose the consumer and provider type for the cluster interface.

**Step 2**
Create the device selection policy.

Specify which cluster interface and bridge domain should be used for each service graph rendering. Service-Graph1 uses the external cluster interface as the consumer connector and the DMZ cluster interface as the provider connector.
Service-Graph2 uses the DMZ cluster interface as the consumer connector and the internal cluster interface as the provider connector.

Configuration Example for a Physical Appliance That is Used By Multiple Service Graphs

The following figure shows a configuration example with the physical Cisco ASA having three interfaces that are used by two service graphs.

*Figure 57: Cisco ASA Physical Device with Three interfaces That are Used by Two Service Graphs*

This example has one consumer endpoint group and two provider endpoint groups.

The following procedure creates the example configuration.

**Procedure**

**Step 1**
Create a Layer 4 to Layer 7 device.
You do not need to add multiple cluster interfaces in the Layer 4 to Layer 7 device because the VLAN trunk is supported on a physical appliance.

**Step 2**
Create the device selection policy.
Use the same cluster interface for both service graphs. However, the bridge domain (BD) for the provider side is different, and so you must create a different sub-interface on the service device.
Service-Graph1 uses the consumer cluster interface as the consumer connector and the provider cluster interface as the provider connector. The provider side is BD2.

Service-Graph2 uses the consumer cluster interface as the consumer connector and the provider cluster interface as the provider connector. The provider side is BD3.

### Verifying the Service Graph Configuration for a Device That is Used By Multiple Service Graphs Using the GUI

After a service graph is deployed successfully, you can see the service graph in the Deployed Devices properties as having multiple cluster interfaces.

**Procedure**

1. **Step 1** On the menu bar, choose Tenants > All Tenants.
2. **Step 2** In the Work pane, double-click the tenant's name.
3. **Step 3** In the Navigation pane, choose Tenant tenant_name > L4-L7 Services > Deployed Devices > device_name.
4. **Step 4** In the Work pane, you can see the properties of the device. The Cluster Interfaces table lists the interfaces.

### Additional References for Reusing a Single Layer 4 to Layer 7 Device for Multiple Service Graphs

For more information about service graphs, including using a single device for multiple service graphs, see the Service Graph Design with Cisco Application Centric Infrastructure White Paper at the following URL:


### Service Graphs with Route Peering

#### About Service Graphs with Route Peering

Route peering is a special case of the more generic Cisco Application Centric Infrastructure (ACI) fabric as a transit use case, in which route peering enables the ACI fabric to serve as a transit domain for Open Shortest Path First (OSPF) or Border Gateway Protocol (BGP) protocols. A common use case for route peering is route health injection, in which the server load balancing virtual IP address is advertised over OSPF or internal BGP (iBGP) to clients that are outside of the ACI fabric. You can use route peering to configure OSPF or BGP peering on a service device so that the device can peer and exchange routes with the ACI leaf switch to which it is connected.
When to Use a Service Graph with Route Peering

In many cases, the service appliance (typically a perimeter firewall) is placed in front of external connectivity, which is an L3Out in ACI. From the ACI perspective, you most likely have a contract between an L3Out and a server-side endpoint group, and you insert the service graph into the contract. In this case, there is a segment for the connector of a service appliance, such as BD1 in the following figure:

Figure 58: L3Out, Contract, and EPG Connection

There are some routing considerations. Traffic is routed based on the destination IP address, as illustrated in the following figure:

Figure 59: Traffic is Routed Based on the Destination IP Address

<table>
<thead>
<tr>
<th>Callout</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Need to know the destination subnet, which is 192.168.2.0/24.</td>
</tr>
</tbody>
</table>

If the Cisco ASA firewall does not do NAT, ACI VRF1 needs to know the 192.168.2.0/24 route. However, if the ACI fabric has subnet 192.168.2.254/24 in BD2, then the traffic from the L3Out will be going directly to the Web server instead of going through the Cisco ASA firewall. As such, you must add a static route or enable dynamic routing between the ACI fabric and Cisco ASA firewall accordingly.
In ACI, use an L3Out to add a static route or enable dynamic routing on the VRF. With an L3Out, you connect the Cisco ASA firewall as an external router in another L3Out (ASA-external). This is one example of when to use a service graph with route peering, which is illustrated in the following figure:

**Figure 60: L3Out Route Peering**

![L3Out Route Peering Diagram]

<table>
<thead>
<tr>
<th>Callout</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Add the 192.168.2.0/24 route. This can be static or dynamic routing.</td>
</tr>
<tr>
<td>2</td>
<td>Route peering on the external side of the Cisco ASA firewall.</td>
</tr>
</tbody>
</table>

Another example of when to use a service graph with route peering is if you want to use an ACI anycast gateway as the default gateway of the servers, as illustrated in the following figure:

**Figure 61: Anycast Gateway as the Default Gateway of the Servers**

![Anycast Gateway Diagram]

<table>
<thead>
<tr>
<th>Callout</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Traffic is not going through the Cisco ASA firewall because the Cisco Application Centric Infrastructure (ACI) fabric in VRF1 knows the 192.168.20.0/24 route as the direct connect route.</td>
</tr>
</tbody>
</table>

If the Cisco ASA firewall does not do NAT, you must use route peering and different VRFs, as illustrated in the following figure:
Prerequisites for Service Graphs with Route Peering

You must understand the basic terminology and configuration for a service graph, L3Outs, and transit routing.

Guidelines and Limitations for Service Graphs with Route Peering

If you use a service graph in managed mode with route peering using the dynamic routing protocol, the device package must be capable of using route peering. The Cisco ASA and Citrix device packages support route peering.

The following dynamic routing protocols are supported:

- OSPF
- OSPF v3
- BGP
- BGP v6

Recommended Configuration Procedure for Service Graphs with Route Peering

The following procedure provides an overview of the steps for configuring a service graph with route peering. For more information about any of the steps, see the Cisco APIC Layer 4 to Layer 7 Services Deployment Guide at the following URL:

Procedure

Step 1  Uploading device package.
Step 2  Create a Layer 4 to Layer 7 device.
Step 3  Create an L3Out for service node connectivity.
Step 4  Create a service graph template.
Step 5  Apply the service graph template.

Configuration Examples for Service Graphs with Route Peering

The procedure in this section provides an overview of configuring a service graph with route peering using an Cisco ASA example. The following figure illustrates the topology this configuration:

*Figure 63: Topology of Configuring a Service Graph to Use Route Peering*

![Topology of Configuring a Service Graph to Use Route Peering](image)

Procedure

Step 1  Create BD1 in VRF1, BD2 in VRF2, and LB BD in VRF2.
Step 2  If you want to use OSPF for route peering, configure the Route Tag policy to use a different tag for VRF1 and VRF2.
Step 3  Upload the device package to the Application Policy Infrastructure Controller (APIC).
Step 4  Set the contract scope to Tenant, unless the service graph is across a tenant, in which case set the contract scope to Global.
Step 5  Create a Layer 4 to Layer 7 device.

With route peering, you must specify the path in the Layer 4 to Layer 7 device even though you are using a virtual appliance. If you do not use route peering with the virtual appliance, the path is not mandatory.

Step 6  Create L3Out ASA-external and ASA-internal for service node connectivity.
The VLAN used in the logical interface profile of the L3Outs will be used for the service node configuration. The APIC and service graph will automatically pick up the VLAN ID routing information and will configure OSPF on the service node.

If you use OSPF, you must configure L3Out subnets accordingly. The subnets are bridge domain subnets that will be advertised to the Cisco ASA firewall when they are marked with the Advertised Externally scope.

**Step 7**
Create the service graph template.

Cisco ASA must be in routed mode and the **Unicast Routes** value should be set to **True**.

**Step 8**
Apply the service graph template.

In the **Apply Service Graph Service Template to EPGs** dialog box, choose where you will place the service node's connectors. When you use route peering, choose the L3Out. If you use a dynamic routing protocol (OSPF or BGP), in the **Router Config** drop-down list, you must choose **Create Router Configuration** to specify the router ID for the service node. In this example, choose **ASA-external** for the consumer and **ASA-internal** for the provider.

**Step 9**
Verify the service graph deployment.

If the service graph is deployed successfully, you can see that the deployed device and Cisco ASA cluster interface use the same VLAN encapsulation with the VLAN ID in the L3Out.

a) In the CLI, check the Cisco ASA OSPF neighbor and routing table:

**Example:**

```
ASA5525X/T1# show ospf neighbor
Neighbor ID Pri State Dead Time Address Interface
11.11.11.11 1 FULL/DR 0:00:36 192.168.2.254 externalIf
13.13.13.13 1 FULL/DR 0:00:34 192.168.1.254 internalIf
```

```
ASA5525X/T1# show route
S* 0.0.0.0 0.0.0.0 [1/0] via 172.16.255.254, management
O E2 10.10.10.0 255.255.255.0 [110/20] via 192.168.1.254, 00:00:32, internalIf
C 172.16.0.0 255.255.0.0 is directly connected, management
L 172.16.0.101 255.255.255.255 is directly connected, internalIf
C 192.168.1.0 255.255.255.0 is directly connected, internalIf
L 192.168.1.101 255.255.255.255 is directly connected, internalIf
C 192.168.2.0 255.255.255.0 is directly connected, internalIf
L 192.168.2.101 255.255.255.255 is directly connected, internalIf
O E2 192.168.20.0 255.255.255.0 [110/20] via 192.168.2.254, 00:00:32, externalIf
```

b) In the CLI, check the leaf routing table (VRF1) to make sure that VRF1 has the 10.10.10.0/24 route.

**Example:**

```
Leaf3# show ip route vrf T1:VRF1
<snip>
1.1.1.1/32, ubest/mbest: 1/0
   *via 192.168.30.1, etht1/21, [110/41], 5d02h, ospf-default, intra
10.10.10.0/24, ubest/mbest: 1/0
   *via 192.168.2.101, vlan20, [110/20], 00:00:27, ospf-default, type-2, tag 200
11.11.11.1/32, ubest/mbest: 2/0, attached, direct
   *via 11.11.11.1, lo3, [1/0], 5d02h, local, local
   *via 11.11.11.1, lo3, [1/0], 5d02h, direct
192.168.1.0/24, ubest/mbest: 1/0
   *via 192.168.2.101, vlan20, [110/14], 00:15:51, ospf-default, intra
192.168.2.0/24, ubest/mbest: 1/0, attached, direct
```
Dynamic Routing Protocol Parameters for OSPF and BGP

The Application Policy Infrastructure Controller (APIC) provides native support for configuring OSPF and BGP parameters for a service node, which implies that a device package does not need to model the OSPF and BGP configuration in the device model, but a device package must have the capability to do a dynamic routing configuration. The OSPF- and BGP-related parameter configured in an L3Out on the APIC is passed to the device script, after which the device script will configure OSPF on the service node.

As an example, an external routed network (L3Out) with OSPF area ID 1 would have the following property values:

Table 14: Example L3Out Property Values

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF check box</td>
<td>Checked</td>
</tr>
<tr>
<td>OSPF Area ID field</td>
<td>0.0.0.1</td>
</tr>
<tr>
<td>OSPF Area Type buttons</td>
<td>Regular area</td>
</tr>
</tbody>
</table>

You can also view the configuration using the CLI:

ASA5525X/T1# show run | b ospf
router ospf 1
router-id 10.10.10.1
network 192.168.1.0 255.255.255.0 area 1
network 192.168.2.0 255.255.255.0 area 1
area 1
log-adj-changes
...

In the device selection policy, you can choose the **Redistribute** option, which is also reflected in the service node if device package supports redistribution.

For more information, see the *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide* at the following URL:


**Additional References for Service Graphs with Route Peering**

For more information about service graphs and route peering, see the *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide*.

For more information about device packages, see the *Cisco APIC Layer 4 to Layer 7 Device Package Development Guide*.

You can find these documents at the following URL:


**The Common Tenant and User Tenants**

**About the Common Tenant and User Tenants**

A tenant is a logical container for application policies that enable an administrator to exercise domain-based access control. A tenant represents a unit of isolation from a policy perspective, but it does not represent a private network. Tenants can represent a customer in a service provider setting, an organization or domain in an enterprise setting, or just a convenient grouping of policies.

The common tenant is provided by the system, but can be configured by the fabric administrator. It contains policies that govern the operation of resources accessible to all tenants, such as firewalls, load balancers, Layer 4 to Layer 7 services, and intrusion detection appliances.

The administrator defines user tenants according to the needs of users. They contain policies that govern the operation of resources, such as applications, databases, web servers, network-attached storage, and virtual machines.

If you have Layer 4 to Layer 7 service devices between endpoint groups in different tenants, you must determine whether to define Layer 4 to Layer 7-related configurations in the common tenant or in a user tenant. This section describes design consideration for service graphs across tenants.
Prerequisites for the Common Tenant and User Tenants

Before you can decide where you will define the Layer 4 to Layer 7-related configurations, you must have basic knowledge about service graphs and inter-tenant contracts.

Guidelines for the Common Tenant and User Tenants

To set up a service graph, you must define the following objects, of which some objects must be visible from the other objects:

<table>
<thead>
<tr>
<th>Object</th>
<th>Consideration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Contract</td>
<td>This object must be visible from the provider and consumer endpoint groups.</td>
</tr>
<tr>
<td>Service graph template</td>
<td>This object must be visible from the contract.</td>
</tr>
<tr>
<td>Layer 4 to Layer 7 device</td>
<td>This object must be visible from the device selection policy.</td>
</tr>
<tr>
<td>Device selection policy</td>
<td>This object must be defined under the provider side endpoint group tenant. This object must be able to see the cluster interfaces in the Layer 4 to Layer 7 device, bridge domains, and L3Out.</td>
</tr>
</tbody>
</table>

Objects defined in the common tenant can be referenced from other tenants, but objects defined in a user tenant can be referenced only from the same tenant. The following examples show that where you define these objects depends on your requirements:

Contract:

- If you want to enable a tenant user to manage the contract filter, the contract must be defined in the provider side endpoint group tenant and the contract must be exported to consumer side endpoint group tenant.
- If you want to hide the security policy from the user tenant, the contract must be defined in the common tenant. The security policy cannot be changed from a user tenant and can be referenced from user tenants without being exported.

Service graph template:

- If your contract is in a user tenant, the service graph template must be defined in the same tenant or the common tenant.
- If your contract is in the common tenant, the service graph template must be in the common tenant.

Layer 4 to Layer 7 device:

- If your provider endpoint group is in a user tenant, it must be defined in the same tenant or exported from another tenant.
- If your provider endpoint group is in the common tenant, it must be defined in the common tenant.

Device selection policy:

- If the device selection policy is in the common tenant, the bridge domain or L3Out for the cluster interface must be in the common tenant.
• If the device selection policy is in a user tenant, the bridge domain or L3Out for the cluster interface must be in the same tenant or the common tenant.

**Example of Where to Define Layer 4 to Layer 7-Related Objects**

This section provides an example of where you must define Layer 4 to Layer 7-related objects. Assume that you have the following requirements:

- You will have a consumer L3Out endpoint group in the common tenant VRF and provider endpoint groups in the user tenant VRF.
- You will use service graph route peering.
- You will define the contract in the user tenant.

*Figure 64: Topology of Requirements*

To meet the requirements, you must define the objects as follows:

- The contract in the user tenant and export it to the common tenant.
- The service graph template in the user tenant or common tenant.
- The Layer 4 to Layer 7 device in the user tenant or exported from another tenant.
- The provider endpoint group, bridge domain, and VRF in the user tenant.
- The consumer endpoint group, bridge domain, and VRF in the common tenant.
- The device selection policy in the user tenant, since the provider side is the user tenant.
- L3Out facing to the Cisco ASA internal side (provider) in the user tenant or common tenant.
- L3Out facing to the Cisco ASA external side (consumer) in the common tenant.
Additional References for the Common Tenant and User Tenants

For more information about service graph design, see the Service Graph Design with Cisco Application Centric Infrastructure White Paper at the following URL:

CHAPTER 7

Miscellaneous Design

- Hardware Choices, on page 139
- Leaf Node Categorization, on page 143
- Fabric Provisioning, on page 144
- About Fabric Provisioning, on page 144

Hardware Choices

About Hardware Choices

Cisco Application Centric Infrastructure (ACI) offers a variety of hardware platforms. Choose a platform based on the type of physical layer connectivity you need, the amount of ternary content-addressable memory (TCAM) space and buffer space you need, and whether you want to use IP-based classification of workloads into endpoint groups (EPGs).

The following table provides a summary of the hardware options that were available for the Application Policy Infrastructure Controller (APIC) 1.3(2f) release. You should refer to the Cisco product page for the most up-to-date information.

Table 15: ACI Fabric Hardware Options

<table>
<thead>
<tr>
<th>Platform</th>
<th>Policy TCAM</th>
<th>IP-based EPGs</th>
</tr>
</thead>
<tbody>
<tr>
<td>9396PX</td>
<td>Regular TCAM with M12PQ</td>
<td>Yes with M6PQ-E</td>
</tr>
<tr>
<td></td>
<td>Bigger TCAM with M6PQ or M6PQ-E</td>
<td></td>
</tr>
<tr>
<td>9396TX</td>
<td>Regular TCAM with M12PQ</td>
<td>Yes with M6PQ-E</td>
</tr>
</tbody>
</table>
### About Hardware Choices

<table>
<thead>
<tr>
<th>Model</th>
<th>Port Count</th>
<th>Host Ports Type</th>
<th>Use (leaf/spine)</th>
<th>Policy TCAM</th>
<th>IP-based EPGs</th>
</tr>
</thead>
<tbody>
<tr>
<td>93128TX</td>
<td>96 x 1/10-Gigabit ports and 8 x 40-Gigabit ports</td>
<td>10GBASE-T</td>
<td>Leaf</td>
<td>Regular TCAM with M12PQ</td>
<td>Yes with M6PQ-E</td>
</tr>
<tr>
<td>9372PX</td>
<td>48 x 1/10-Gigabit ports and 6 x 40-Gigabit ports</td>
<td>10-Gigabit SFP+</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>No</td>
</tr>
<tr>
<td>9372TX</td>
<td>48 x 1/10-Gigabit ports and 6 x 40-Gigabit ports</td>
<td>10GBASE-T</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>No</td>
</tr>
<tr>
<td>93108TC-EX</td>
<td>96 x 1/10-Gigabit ports and 6 x 100-Gigabit ports</td>
<td>100-Gigabit QSFP28</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>No</td>
</tr>
<tr>
<td>93120TX</td>
<td>96 x 1/10-Gigabit ports and 6 x 40-Gigabit ports</td>
<td>10GBASE-T</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>No</td>
</tr>
<tr>
<td>93180YC-EX</td>
<td>48 x 10/25-Gigabit ports and 6 x 40/100-Gigabit ports</td>
<td>40-Gigabit QSFP28</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>No</td>
</tr>
<tr>
<td>9332PQ</td>
<td>32 x 40-Gigabit ports</td>
<td>40-Gigabit QSFP+</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>No</td>
</tr>
<tr>
<td>9372PX-E</td>
<td>48 x 1/10-Gigabit ports and 6 x 40-Gigabit ports</td>
<td>10-Gigabit SFP+</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
<td>Yes</td>
</tr>
</tbody>
</table>
## Expansion Modules

You can choose among three expansion modules according to the switches you are using and your needs:

- **Cisco M12PQ**—Twelve 40-Gbps ports with an additional 40 MB of buffer space and a smaller TCAM compared to the other models. It can be used with the Cisco Nexus 9396PX, 9396TX, and 93128TX switches.

- **Cisco M6PQ**—Six 40-Gbps ports with additional policy TCAM space. It can be used with the Cisco Nexus 9396PX, 9396TX, and 93128TX switches.

- **Cisco M6PQ-E**—Six 40-Gbps ports with additional policy TCAM space. It can be used with the Cisco Nexus 9396PX, 9396TX, and 93128TX switches and allows you to classify workloads into EPGs based on the IP address of the originating workload.

## Leaf Switches

In the ACI, all workloads connect to leaf switches. The leaf switches used in an ACI fabric are ToR switches. They are divided into four main types based on their hardware:

<table>
<thead>
<tr>
<th>Port Count</th>
<th>Host Ports Type</th>
<th>Use (leaf/spine)</th>
<th>Policy TCAM</th>
<th>IP-based EPGs</th>
</tr>
</thead>
<tbody>
<tr>
<td>9372TX-E</td>
<td>48 x 1/10-Gigabit ports and 6 x 40-Gigabit ports</td>
<td>10GBASE-T</td>
<td>Leaf</td>
<td>Bigger TCAM</td>
</tr>
<tr>
<td>9336PQ</td>
<td>36 x 40-Gigabit ports</td>
<td>40-Gigabit QSFP+</td>
<td>Spine</td>
<td>N/A</td>
</tr>
<tr>
<td>9504</td>
<td>With 9736PQ: 36 x 40-Gigabit ports per linecard</td>
<td>40-Gigabit QSFP+</td>
<td>Spine</td>
<td>N/A</td>
</tr>
<tr>
<td>9508</td>
<td>With 9736PQ: 36 x 40-Gigabit ports per linecard</td>
<td>40-Gigabit QSFP+</td>
<td>Spine</td>
<td>N/A</td>
</tr>
<tr>
<td>9516</td>
<td>With 9736PQ: 36 x 40-Gigabit ports per linecard</td>
<td>40-Gigabit QSFP+</td>
<td>Spine</td>
<td>N/A</td>
</tr>
</tbody>
</table>
• Border Leaf—The border leaf switches are ACI leaf switches that provide Layer 2 or Layer 3 external connectivity to outside networks. The border leaf supports routing protocols to exchange routes with external routers, and it also applies and enforces policies for traffic between internal and external endpoints.

• Service Leaf—The service leaf switches are ACI leaf switches that connect to Layer 4-7 service appliances, such as firewall, load balancer, and such. The connectivity between the service leaf and the service appliance can be Layer 2 or Layer 3 depending on design scenarios.

• Compute Leaf—The compute leaf switches are ACI leaf switches that connect to compute systems. The compute leaf supports individual port, port channel, and virtual port channel (vPC) interfaces, based on the nature and requirements of the application or the system. It also applies and enforces policies for traffic to and from local endpoints.

• IP Storage Leaf—The storage leaf switches are ACI leaf switches that connect to IP storage systems. It supports individual port, port channel, and virtual port channel (vPC) interfaces based on the nature and requirements of the application and the system. It also applies and enforces policies for traffic to and from local endpoints.

While it is not a requirement to have dedicated switches to serve certain functions, it is preferred especially for a large data center.

It is easier to standardize configuration templates and enables applications to flexibly tap into any available resources.

For example, a large data center that supports high volume of traffic between the ACI fabric and the core network, might choose to designate two border leaf switches for high availability and scalability considerations.

### Spine Switches

The Cisco ACI fabric forwards traffic primarily based on host lookups. A mapping database stores the information about the ToR switch on which each IP address resides. This information is stored in the fabric cards of the spine switches.

The spine switches have several form factors. The models also differ in the number of endpoints that they can hold in the mapping database, which depends on the number of fabric modules installed. Modular switches equipped with six fabric modules can hold the following numbers of endpoints:

- Fixed form-factor Cisco Nexus 9336PQ—Up to 200,000 endpoints
- Modular 4-slot switch—Up to 300,000 endpoints
- Modular 8-slot switch—Up to 600,000 endpoints
- Modular 16-slot switch—Up to 1.2 million endpoints

You can mix spine switches of different types, but the total number of endpoints that the fabric supports is the minimum common denominator. You should stay within the maximum tested limits for the software, which are shown in the Capacity Dashboard in the APIC GUI. At the time of this writing, the maximum number of endpoints that can be used in the fabric is 180,000.

Also keep in mind when choosing the platform:

- Allow for future growth.
• Verify that the features that you want to deploy are supported on the selected platform. For example, the IP-based EPG feature requires the -E, -EX, or later versions of leaf switches.

• Make sure the leaf switch TCAM size is large enough to support the contracts or application rules that will be deployed within the fabric.

• When using two leaf switches for a vPC pair, make sure to use the same switch model to avoid any corner issues.

• Use two or more spine switches for higher bandwidth and for redundant connections to external networks.

Additional References for Hardware Choices

For more information about hardware choices, see:

Leaf Node Categorization

About Leaf Node Categorization

When deploying an Cisco Application Centric Infrastructure (ACI) fabric, you usually delegate specific devices and services to specific leaf nodes. This enables you to understand quickly where issues might be located, given the state of the leaf nodes. This also enables fast diagnosis for node troubleshooting. Typically, the special-purpose leaf categories are as follows:

• Border leaf
• Compute Leaf
• Services Leaf
• Storage Leaf

Prerequisites for Leaf Node Categorization

The following are the prerequisites for leaf node categorization:

• Understand the appliances and devices to be added to the fabric.
• Understand the design to be implemented in the fabric.

Guidelines and Limitations for Leaf Node Categorization

Leaf node categorization enables a network operator to easily distinguish the purposes of leaf nodes when they have issues, whether in troubleshooting or further implementation and growth. There is no strict definition of the categories to be used, nor is there a configuration on the Cisco Application Centric Infrastructure (ACI) fabric to enforce these categories. The categories are only a set of labels that are typically used in the ACI fabric.
• Border Leaf—This leaf node is typically connected to L3 Outs. L3 Outs can serve as a path into the WAN, or into the core of a legacy network.

• Compute Leaf—This leaf node is typically connected to compute resources, whether the resources are physical or virtualized servers.

• Services Leaf—Services within ACI are typically those given by Layer 4 to Layer 7 services. Services include firewalls, load balancers, and intrusion prevention systems. Services do not need to be integrated into ACI through a service graph template to be considered a service; that is a definition from the applications point of view.

• Storage Leaf—This leaf node is typically connected to storage devices for compute resources. This can include iSCSI, NFS, or other Ethernet medium storage devices.

Leaf nodes do not need to be delegated to only one category. Depending on the design, the categories can overlap. For example, a leaf node serving as a border leaf node can also provide compute resources.

**Additional References for Leaf Node Categorization**

For additional information on border leaf switches:


**Fabric Provisioning**

**About Fabric Provisioning**

**Fabric Infrastructure IP Range Recommendations**

When provisioning an Application Policy Infrastructure Controller (APIC), one of the required data points during the setup stage is an IP address range for infrastructure addressing inside of the fabric. This is primarily for the purposes of allocating tunnel endpoint (TEP) addresses. The default value for this range is 10.0.0.0/16. Although technically you can select a range that overlaps with other subnets in the network, you should choose a unique range for this infrastructure range.

Frequently, the infrastructure IP address range must be extended beyond the Cisco Application Centric Infrastructure (ACI) fabric. For example, when the Application Virtual Switch (AVS) is used, a VMK interface is automatically created that uses an address from the infrastructure range as shown in the following figure:
If the infrastructure range overlaps with other subnets elsewhere in the network, routing problems might occur.

The minimum supported subnet size in the recommended three APIC scenario is /22. The number of addresses required depends on a variety of factors, including the number of APICs in your fabric, the number of leaf and spine nodes, the number of AVS instances, and the number of virtual port channels required. To avoid issues with address exhaustion, you should consider allocating a /16 or /17 range if possible.

When considering the preceding requirements, remember that changing either the infrastructure IP address range or the VLAN after initial provisioning is not possible without rebuilding the fabric.

**Fabric Infrastructure VLAN Recommendations**

During fabric provisioning, the system requires a VLAN number to be used as the infrastructure VLAN. This VLAN is used for control communication as a reserved overlay VLAN between the fabric nodes (leaf, spine, and APIC controllers) to bring up the fabric. This VLAN is hard coded on the fabric nodes.

If possible, this VLAN number should be unique within the network. In a scenario where the infrastructure VLAN is extended outside of the ACI fabric (for example, if using Cisco AVS or OpenStack integration with Opflex), this VLAN might need to traverse other (non-ACI) devices. In that case, be sure that the infrastructure VLAN does not fall within a range that is prohibited on the non-ACI device. The following figure shows an example of the reserved VLAN range within a Cisco Nexus 7000:
In many cases, VLAN 3967 is a good choice for the ACI infrastructure VLAN to avoid the issue outlined in the preceding section.

For more information about fabric infrastructure VLAN recommendations, see the Cisco APIC Getting Started Guide at the following URL:


**Fabric Node ID Recommendation**

The fabric node ID is used to form the fabric membership during fabric initialization. It is also used to configure underlay physical policies, such as access policies and fabric polices, within the fabric. Having a good node ID structure is important to ease the management and operation for the ACI fabric.

Below are general guidelines for configuring fabric node IDs:

- Plan the node ID wisely to allow for further growth and expansion.
- Use different node ID ranges for spine switches and leaf switches. For example, the 100 range for spine switches and the 200 range for leaf switches.
- Using different node ID ranges for leaf switches depends on the use case. For example, if leaf switches are categorized in a different functionality, consider using a different range based on the use. For example, the 200 range for border leaf switches and service leaf switches, the 300 range for compute leaf switches, and the 400 range for storage leaf switches.
Node IDs 1 through 29 are reserved for APICs, which cannot be changed. When APIC redundancy is configured, you should use IDs 1 to 19 for active APICs and IDs 20 to 29 for standby APICs. This allows for expansion of the fabric.

- When a pair of switches is used for the server uplink connectivity using either vPC or active/standby, consider using sequential numbers for the leaf node ID for those switch pairs. For example, node ID 201 for the vPC side A connectivity and node ID 202 for side B. That way, it is easier to configure and easier to manage an upgrade when using maintenance groups.

- If only one ToR switch is deployed, reserve the even leaf ID for future use.

Once the fabric node ID is assigned, the ID is difficult to change unless the fabric nodes (spine and leaf) are decommissioned from the fabric and cleanly rebooted.
PART II

Implementation

• ACI Constructs Implementation, on page 151
• Routing Implementation, on page 167
• Virtualization Implementation, on page 175
• Miscellaneous Implementation, on page 183
ACI Constructs Implementation

• Configuration Zones, on page 151
• Shared Services, on page 153
• EPG Static Binding, on page 155
• In-Band and Out-of-Band Management, on page 158
• Out-of-Band Management Contracts, on page 161

Configuration Zones

About Configuration Zones

Configuration zones divide the Cisco Application Centric Infrastructure (ACI) fabric into different zones that can be updated with configuration changes at different times. This limits the risk of deploying a faulty configuration on the entire fabric at once that might disrupt traffic or even bring the fabric down. An administrator can deploy a configuration to a defined non-critical zone, and then deploy it to defined critical zones when satisfied that it is suitable. Similar to the way that UCS Manager functions, a configuration zone is essentially an additional "user acknowledge" type of policy that forces users to verify configuration changes before applying the changes.

You can choose one of the following deployment modes for a configuration zone:

• Enabled—Pending updates are sent immediately
• Disabled—New updates are postponed
• Triggered—Pending updates are sent immediately, and the deployment mode is reset to the value it had before being triggered

Without configuration zones enabled, policy changes will take effect on all fabric nodes once the configuration is set and standard programming criteria are met. With configuration zones enabled, you can have these policy changes transition to a state of "postponed" until a user acknowledges the change to be applied in specific zones.

Zones can encompass an entire POD, or can encompass a subset of fabric nodes.
Prerequisites for Configuration Zones

You must meet the following prerequisites to use configuration zones:

- You must be using the Application Policy Infrastructure Controller (APIC) 1.2(2) release or later

Guidelines and Limitations for Configuration Zones

The following guidelines and limitations apply for configuration zones:

- Do not upgrade, downgrade, commission, or decommission nodes in a disabled configuration zone.
- Nodes can only be part of a single zone. Attempting to place a node in multiple zones will generate a server error.
- Do not separate virtual port channel (vPC) member nodes into different configuration zones. If the nodes are in different configuration zones, then the vPCs' modes become mismatched if the interface policies are modified and deployed to only one of the vPC member nodes.

Recommended Configuration procedure for Configuration Zones

As configuration zones can be manually defined, zone definition will typically encompass a logical "non-critical zone" and a "critical zone." The intent is that fabric-wide changes can be allowed to be made only on the non-critical zone first. This gives network operators a chance to verify the configuration and behaviors to ensure expectations are met. Once verification has been performed on the non-critical zone, the change can then be applied to the critical zone.

For procedures for defining configuration zones, see the Cisco APIC Troubleshooting Guide at the following URL:


Verifying the Configuration Zones Using the GUI

The following procedure verifies the configuration zones by using the Application Policy Infrastructure Controller (APIC) GUI.

Procedure

<table>
<thead>
<tr>
<th>Step 1</th>
<th>On the menu bar, choose <strong>System &gt; Config Zones</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the <strong>Work</strong> pane, in the <strong>Select Zone</strong> drop-down list, choose the zone that you want to verify.</td>
</tr>
<tr>
<td></td>
<td>You can view and modify the zone's configuration.</td>
</tr>
</tbody>
</table>
Configuration Examples for Configuration Zones

For configuration zone examples using the different user interfaces, see the *Cisco APIC Configuration Zones* knowledge base article at the following URL:


Additional References for Configuration Zones

For more information about configuration zones, see the *Cisco APIC Configuration Zones* knowledge base article at the following URL:


Shared Services

About Shared Services

Shared services is the paradigm of taking endpoints within one tenant/VRF and allowing them to communicate with endpoints within another tenant/VRF. Shared services enables this communications across tenants while preserving the isolation and security policies of the individual tenants. A routed connection to an external network is an example of a shared service that multiple tenants use.

Prerequisites for Shared Services

You must meet the following prerequisites to use shared services:

• A Cisco Application Centric Infrastructure (ACI) fabric that has been fully initialized

• At least 2 user-created tenants to share services between, one will be the provider and one will be the consumer

• At least 1 EPG within each of these tenants

• A subnet defined under the provider EPG as “Shared between VRFs”

Guidelines and Limitations for Shared Services

The following guidelines and limitations apply when using shared services:

• As of Release 1.2, shared services can be performed with a shared subnet defined under the bridge domain (BD).
  • The preferred method remains having the shared subnet defined under the EPG to be shared to another tenant.
  • Prior to this, shared services required that the provider subnet be defined under the EPG that was to be shared.
- Contracts for shared service must have the scope set to **Global**. The default scope is **VRF** and will not work for shared services.

- For BD-to-BD shared services: Given User-Tenant A to User-Tenant B, each tenant has a contract that is associated as a provider under an EPG and is exported to the other tenant. The same EPG takes the subsequently imported contract and has it applied as a consumed contract interface.
  - All EPGs that communicate with BD to BD Shared Services have at least two contract relationships, one as a provider and one as a consumed contract interface.

- When using BD-to-BD shared services, due to the extra configuration and rules associated with having a provider set within both tenants, limit the fabric to roughly 16k EPGs.

- In the case of vzAny, you must define the provider EPG shared subnet under the EPG in order to properly derive the pCTag (classification) of the destination from the consumer (vzAny) side. If you are migrating from a BD-to-BD shared services configuration, where both the consumer and provider subnets are defined under bridge domains, to vzAny acting as a shared service consumer, you must take an extra configuration step where you add the provider subnet to the EPG with the shared flags at minimum.

  **Note**
  If you add the EPG subnet as a duplicate of the defined BD subnet, ensure that both definitions of the subnet always have the same flags defined. Failure to do so can result in unexpected fabric forwarding behavior.

- Subnets leaked from multiple consumer networks into a VRF, or vice versa, must be disjointed and must not overlap. If two consumers are mistakenly configured with the same subnet, recovery from this condition is done by removing the subnet configuration for both then reconfiguring the subnets correctly.

- Subnets leaked across VRFs must have the **Shared between VRFs** and **ND RA Prefix** options enabled, to be defined on the BD or the EPG.

**Recommended Configuration Procedure of Shared Services Using the GUI**

The following procedure to configure the contract(s) that will utilize shared services using the Application Policy Infrastructure Controller (APIC) GUI.

**Procedure**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>From the menu bar, choose <strong>Tenants</strong> &gt; <strong>tenant_name</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the <strong>Navigation</strong> pane, choose <strong>Security Policies</strong> &gt; <strong>Contracts</strong> &gt; <strong>contract_name</strong>.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the <strong>Work</strong> pane, click the <strong>Policy</strong> tab and set the <strong>Scope</strong> field to <strong>Global</strong>.</td>
</tr>
</tbody>
</table>

**Note**
If performing BD-BD shared services, a contract set to scope **Global** should exist within both tenants to be exported to one another.
Configuration Examples for Shared Services Using the GUI

The following procedure provides an example of configuring shared services using the Application Policy Infrastructure Controller (APIC) GUI.

**Procedure**

**Step 1**
To set the shared services contract as a provider for an EPG with a shared subnet: on the menu bar, choose Tenants > tenant_name.

**Step 2**
In the Navigation pane, choose tenant_name > Application Profiles > profile_name > Application EPGs > epg_name > Contracts.

**Step 3**
Right-click on Contracts, choose Add Provided Contract, and enter a name for the contract in the Name field.

**Step 4**
To export the contract from one tenant to another: in the Navigation pane, choose Security Profiles > Contracts.

a) Right-click on Contracts and choose Export Contract. Enter the appropriate information for the Name, Contract and Tenant fields. Click Submit when finished.

**Step 5**
To apply the contract to the consumer EPG within the imported tenant as a consumed contract interface: in the Navigation pane, choose tenant_name > Application Profiles > profile_name > Application EPGs > epg_name > Contracts.

**Step 6**
Right-click on Contracts, choose Add Consumed Contract Interface, and enter a name for the contract in the Name field.

**Note**
If performing BD-BD shared services, repeat the procedure between tenants before communication will be successful between both EPGs.

Additional References for Shared Services

For more information on EPG Static Binding Modes, see the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:


EPG Static Binding

About EPG Static Binding Modes

Static bindings enable you to statically link the EPG to either a path (node/interface) or to an entire leaf. This binding essentially forces the bound node to perform programming of the defined VLAN for classification of incoming traffic. Without a static binding, traffic going into the fabric will not be classified into an EPG and subsequently will not be forwarded.
Prerequisites for EPG Static Binding Modes

You must meet the following prerequisites to use EPG static binding modes:

- The Cisco Application Centric Infrastructure (ACI) fabric must be initialized.
- Access policies must be configured that correspond to the defined path.

Guidelines and Limitations for EPG Static Binding Modes

The following guidelines and limitations apply when using EPG static binding mode:

- If access policies associated with a domain have not been provisioned properly, the EPG will generate a fault when a static binding is applied.
- Faults indicating invalid path typically refer to some missing access policies given the defined path.
- Faults indicating VLAN issues typically refer to a missing VLAN association given the defined path.
- When a port is set to Untagged, that port can no longer be utilized as an untagged port in other EPGs.
  - For this to be accomplished, deploy the EPG instead as 802.1p.
- When utilizing 802.1p defined ports with other definitions on the same port as trunked, packets will egress this interface as VLAN-0, or as untagged in the case of EX switches.
  - Most devices process VLAN-0 as an untagged packet and have no issues.
  - For hosts that cannot VLAN-0 as an untagged packet, the setting must be Untagged.

Recommended Configuration procedure of EPG Static Binding Modes

The following 3 port modes can be applied when configuring EPG static binding modes:

- **Trunk (Tagged - classic IEEE 802.1q trunk)**—Traffic for the EPG is sourced by the leaf switch with the specified VLAN tag. The leaf switch also expects to receive traffic tagged with that VLAN to be able to associate it with the EPG. Traffic received untagged is discarded.
- **Access (Untagged)**—Traffic for the EPG is sourced by the leaf as untagged. Traffic received by the leaf switch as untagged or with the tag specified during the static binding configuration is associated with the EPG.
- **Access (802.1p)**—If only one EPG is bound to that interface, the behavior is identical as in the untagged case. If other EPGs are associated with the same interface, traffic for the EPG is sourced with an IEEE 802.1q tag using VLAN 0 (IEEE 802.1p tag), or is sourced as untagged in the case of EX switches.

Verifying the EPG Static Binding Modes Using the GUI

The following procedure verifies the EPG static binding mode configuration using the Cisco Application Policy Infrastructure Controller (APIC) GUI.
Procedure

Step 1
On the menu bar, choose TENANTS > All Tenants.

Step 2
In the Work pane, double-click the desired tenant’s name.

• If you are using the Advanced GUI Mode of the Cisco APIC GUI, then from the Navigation pane, expand Application Profiles > profile_name > Application EPGs > application_epg_name > Static Ports.

• If you are using the Basic GUI Mode of the Cisco APIC GUI, then from the Navigation pane, expand tenant_name > Application Profiles > profile_name > Application EPGs > application_epg_name.

Step 3
From the Navigation pane, click Static Ports.

Your static ports are listed in a summary table inside the Work pane. See the Mode column in the summary table to verify the EPG static binding modes.

Configuration Examples for EPG Static Binding Modes Using the GUI

The following procedure provides an example of configuring EPG static binding modes using the Application Policy Infrastructure Controller (APIC) GUI.

Procedure

Step 1
Configure contract labels (consumer and provider). On the menu bar, choose TENANTS > All Tenants.

Step 2
In the Work pane, double-click the desired tenant’s name.

• If you are using the Advanced GUI Mode of the APIC GUI, then from the Navigation pane, expand Application Profiles > profile_name > Application EPGs > application_epg_name.

• If you are using the Basic GUI Mode of the APIC GUI, then from the Navigation pane, expand tenant_name > Application Profiles > profile_name > Application EPGs > application_epg_name.

Step 3
In the Navigation pane, right-click on Static Ports to open the Deploy Static EPG On PC, VPC, Or Interface dialog box and perform the following tasks:

a) In the Path Type and Path fields, click the port type and the drop-down menu to navigate the node path.

b) In the Port Encap field, enter in the VLAN ID.

c) In the Deployment Immediacy field, choose the deployment type.

d) In the Mode field, choose the mode type.

e) Click Submit.

Additional References for EPG Static Binding Modes

For more information on EPG static binding modes, see the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:
In-Band and Out-of-Band Management

About In-Band and Out-of-Band Management

You can use in-band or out-of-band when designing the management plane and connectivity for the Cisco Application Centric Infrastructure (ACI) fabric. Out-of-band management utilizes its own set of specific ports that only exist on the out-of-band management plane. There is no configuration available to merge the out-of-band management plane into the data plane of the ACI fabric. Out-of-band management typically has its specific ports connected to a device that only manages out-of-band network traffic. The following figure illustrates out-of-band management:

*Figure 68: Out-of-Band Management*

In-band management refers to utilizing the data plane for management traffic. In the case of ACI, this refers to having Application Policy Infrastructure Controller (APIC)-sourced management ports go through the leaf nodes to allow for management communication to devices hanging directly off of these leaf switch ports. The following figure illustrates in-band management:
Prerequisites for In-Band and Out-of-Band Management

You must meet the following prerequisites to use in-band or out-of-band management:

- Have an understanding of level of tenancy for the environment in question
- Have an understanding of services requiring management communication to the Application Policy Infrastructure Controller (APIC), such as managed Layer 4 to Layer 7 devices or VMM integration
- Have an understanding of potential tenants' management design and how they will present their management network to the Cisco Application Centric Infrastructure (ACI) fabric

Guidelines and Limitations for In-Band and Out-of-Band Management

The following guidelines and limitations apply for in-band and out-of-band management:

- Out-of-band management ports are mgmt0 ports on the switch nodes and the two LAN-On-Motherboard (LOM) ports on the Application Policy Infrastructure Controller (APIC). This configuration should not be changed on the APICs.
• In-band management ports are the front panel ports on the leaf nodes and the two PCIE VIC ports connected to the fabric on the APIC.

• Out-of-band and in-band management connectivity policies reside within tenant "mgmt."

• The out-of-band management address assignment that is set during the APIC startup script does not have an object created to represent that assignment. This must be done after fabric initialization to get an object representation within the MIT.

• The APIC management address sources traffic to the management address of various devices for integrations. For example, the APIC management must have communication to the management address of vCenter for VMM integration to be successful. This can be through in-band or out-of-band.

• When in-band management is set up, the APIC always prefers in-band for any traffic sourced from the APIC. Out-of-band is still accessible for devices that are sending requests to the out-of-band address specifically.

• There is no configuration available to leak the out-of-band management plane from the APIC into the data plane. This can only be accomplished by physically cabling out-of-band network devices directly into the data plane. Cisco does not recommend this setup. The preferred setup for this type of design would be to utilize in-band management.

• When utilizing in-band management with multi-tenancy, shared services will be used extensively to leak tenant management subnets into the fabric's in-band subnet.

Recommended Configuration procedure of In-Band and Out-of-Band Management

For the configuration procedures for in-band and out-of-band management, see the Cisco APIC Basic Configuration Guide at the following URL:


Verifying the In-Band and Out-of-Band Management Configuration Using the GUI

Depending on how the management address was set, there are a few locations from the GUI where you can verify the address assignment. The following procedure shows how to verify the address assignment from the different locations.

Procedure

Step 1 On the menu bar, choose Tenants > mgmt.

Step 2 In the Navigation pane, choose Tenant mgmt > Node Management Addresses > Static Node Management Addresses.

In the Work pane, you can see the in-band and out-of-band static management address assignment.

Step 3 In the Navigation pane, choose Tenant mgmt > Node Management Addresses > name_of_policy.
In the Work pane, you can see the dynamic address assignments that can be created to provision mgmt addresses. If created, they specify the node ID, address assignment, and in-band or out-of-band assignment of the addresses.

Verifying the In-Band and Out-of-Band Management Configuration Using the NX-OS-Style CLI

The following procedure verifies the in-band and out-of-band management configuration using the NX-OS-style CLI.

**Procedure**

**Step 1** View the out-of-band interfaces:

```
apic1# ifconfig oobmgmt
```

**Step 2** View the in-band interfaces:

```
apic1# ifconfig bond 0.vlan
```

`vlan` is the ID of the VLAN that is assigned as the in-band VLAN.

Additional References for In-Band and Out-of-Band Management

For more information about shared services guidelines, see the *Cisco Application Centric Infrastructure Fundamentals Guide*.

For more information about NTP utilizing in-band or out-of-band management, see the *Cisco APIC Basic Configuration Guide*.

You can find these documents at the following URL:


Out-of-Band Management Contracts

About Out-of-Band Management Contracts

For out-of-band management, hosts defined within the external management network instance profile can communicate with the nodes in the out-of-band management endpoint group only by using special out-of-band contracts. Regular contracts cannot be used with the out-of-band management endpoint group.
Prerequisites for Out-of-Band Management Contracts

You must meet the following prerequisites to use out-of-band management contracts:

- The Cisco Application Policy Infrastructure Controllers (APICs) must be setup.
- The fabric must be initialized.

Guidelines and Limitations for Out-of-Band Management Contracts

The following guidelines and limitations apply when using out-of-band management contracts:

- Starting with Cisco APIC Release 1.2(2), when a contract is provided on an out-of-band node management endpoint group, the default Cisco APIC out-of-band contract source address is the local subnet that is configured on the out-of-band node management address. Prior to Cisco APIC Release 1.2(2), any address was allowed to be the default Cisco APIC out-of-band contract source address.

- If a contract is consumed on the external management network instance profile, any flow that is not defined will default in only the out-of-band subnet having access to it.

Recommended Configuration Procedure of Out-of-Band Management Contracts

Using the GUI

The following procedure restricts out-of-band management through contract and subnet definitions within the node management EPG and external management network connectivity profile using the Cisco APIC GUI.

Procedure

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>Configure out-of-band management. On the menu bar, choose Tenants &gt; mgmt.</td>
</tr>
<tr>
<td>Step 2</td>
<td>In the Navigation pane, choose Tenant mgmt &gt; Node Management EPGs &gt; .</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the Work pane, double-click Out-of-Band_name and expand the Provided Out-of-Band Contract table to configure.</td>
</tr>
<tr>
<td>Step 4</td>
<td>Configure the consumer contract association and consumer subnet. In the Navigation pane, choose External Management Network Instance Profiles.</td>
</tr>
<tr>
<td>Step 5</td>
<td>In the Work pane, double-click External Management Network Instance Profile_name and expand the Consumed Out-of-Band Contracts and Subnets tables to configure.</td>
</tr>
</tbody>
</table>

Verifying the Out-of-Band Management Contracts

From the Cisco APIC, out-of-band contracts can be verified from the iptables where a new target entry named fp-default now exists.
**Procedure**

In the chain for **fp-default**, there are entries based on the defined subnet. In the example below, only subnet 192.168.1.0/24 is allowing oob access.

**Example:**

```
pod3-apic1# iptables -L
Chain INPUT (policy DROP)
    target     prot opt source           destination
fp-default  all --  anywhere          anywhere
apic-default-drop all --  anywhere     anywhere
apic-default-allow all --  anywhere    anywhere
apic-default all --  anywhere          anywhere
...
```

Where a new target entry named **fp-default** exists, there are entries based on the defined subnet at the chain for **fp-default**. In the above example, only subnet 192.168.1.0/24 is being allowed out-of-band access.

**Configuration Examples for Out-of-Band Management Contracts**

The following procedure provides an example of configuring out-of-band management contracts using the Cisco APIC GUI.

To establish management connectivity to a Cisco ACI-mode fabric switch or an APIC controller, you must perform the following configuration in APIC.

- Create a node management EPG, either inband or out-of-band, that will include the nodes to be managed (leaf and spine switches and APIC controllers).
- Create an external management network instance profile that will include management hosts.
- Configure and associate a filter and contract to allow communication between the external management network instance profile and the node management EPG.
- Access the **Advanced GUI** mode.
Procedure

**Step 1** On the menu bar, choose TENANTS > mgmt.

**Step 2** In the Navigation pane, expand Security Policies.

**Step 3** Right-click **Out-of-Band Contracts** then click **Create Out-of-Band Contract**.

Regular contracts cannot be used with an out-of-band node management endpoint group.

**Step 4** In the Create Out-of-Band Contracts dialog box, perform the following tasks:

a) In the Name field, enter a name for the contract.

b) Expand **Subjects**. In the Create Contract Subject dialog box, in the Name field, enter a subject name.

b) Expand **Filters**, and in the Name field from the drop-down list, choose the name of the filter (default). Click Update and click OK.

d) In the Create Out-of-Band Contract dialog box, click Submit.

**Step 5** Right-click **Node Management EPGs** and click **Create Out-of-Band Management EPG**.

An out-of-band management endpoint group consists of switches (leaves/spines) and Cisco APICs that are part of the associated out-of-band management zone.

**Step 6** In the Create Out-of-Band Management EPG dialog box, perform the following tasks:

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

b) Expand **Provided Out-of-Band Contracts**, and in the OOB Contract field, from the drop-down list, choose the name of the contract you created. Click Update, and click OK.

The out-of-band contract is associated with the node management EPG.

c) In the Create Out-of-Band Management EPG dialog box, click Submit.

**Step 7** Right-click **External Management Network Instance Profiles** and click **Create External Management Network Instance Profile**.

Hosts that are part of regular endpoint groups cannot communicate with the nodes in the out-of-band management endpoint group. Any host that is part of a special group known as the instance profile can communicate with the nodes in an out-of-band management endpoint group using special out-of-band contracts.

**Step 8** In the Create External Management Network Instance Profile dialog box, perform the following tasks:

a) In the Name field, enter a name for the instance profile.

b) Expand **Consumed Out-of-Band Contracts**, and in the Out-of-Band Contract field, from the drop-down list, choose the name of the contract you created. Click Update.

c) Expand **Subnets** and type the external subnet IP address and subnet mask of the managing hosts. Click Update, and click OK.

The out-of-band contract is associated with the subnet.

d) In the Create External Management Network Instance Profile dialog box, click Submit.

---

**Additional References for Out-of-Band Management Contracts**

For more information on out-of-band management, see the *Cisco Application Centric Infrastructure Fundamentals Guide* at the following URL:
application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html
CHAPTER 9

Routing Implementation

- L3Out Subnets, on page 167

L3Out Subnets

About Defining L3Out Subnets

L3Outs are the Cisco Application Centric Infrastructure (ACI) objects used to provide external connectivity in external Layer 3 networks. The L3Out is where you configure the interfaces, protocols, and protocol parameters that are used to provide IP connectivity to external routers. The following list contains the different managed objects configured under the L3Out.

- Export Route Control Subnet—Controls which external networks are advertised out of the fabric using route-maps and IP prefix-lists.

- External Subnets for External EPG—Classifier for the external EPG. The rules and contracts defined in this external EPG apply to networks matching this subnet.

- L3Outside—Top object for the L3Outside connection. This is where protocol selection (BGP, OSPF, or EIGRP) is done. OSPF area and area definition (regular, nssa, or stub area) and area cost is configured here. EIGRP autonomous system is configured here. VRF selection and external domain is assigned at the L3Out.

- Logical Interface Profiles—The interface configuration for the L3Out is configured. This is the IP address configuration, VLAN configuration, MTU configuration.

- Logical Node Profiles—Node profiles are configured under the logical node profile. This is where the leaf switch selection, router-id, and static route configuration is performed. When and L3Out spans multiple leaf switches, all nodes can be configured under one node profile.

- Networks (L3Out Network Instance Profile)—The external EPG configuration for the L3Out. This where the routing controls, EPG classification, and contract configuration is done here. There can be multiple externals per L3Out and assigned to different contracts.

- Match Rules for Route Maps—L3Outs in ACI support route-map configuration. This section is where route-map match statements are configured.
• **Protocol Policies**—Routing protocol policies are configured here. Policies include interface policies (timers, OSPF network type, passive interface, BFD policies, route summarization policies, and protocol knobs are configured here).

**Note**
L3Outs across different tenants will use similar protocol policies. For example, many OSPF L3Outs may use the same network type or all EIGRP L3Outs may use default interface settings. If protocol policies are defined under the common tenant, all other tenants can use them. This eliminates having to configure the same policies across all tenants.

• **Set Rules for Route Maps**—Route-map set statements are configured. Route map set statements are used to influence routing decisions. Set statements include BGP communities, local preference, weight, route dampening, MED, OSPF metric, and metric type.

• **Shared Route-Control Subnet**—Controls which external prefixes are advertised to other tenants for shared services.

• **Shared Security-Import Subnet**—Configures the classifier for the subnets in the VRF where the routes are leaked.

## Prerequisites for Defining L3Out Subnets

You must meet the following prerequisite before defining L3Out subnets:

• **BGP Route Reflector Policy**—L3Outs are used to provide connectivity to external Layer 3 networks. Whenever L3Outs are configured, the BGP route reflector policy should be configured to propagate external routes within the Cisco ACI fabric.

## Guidelines and Limitations for Defining L3Out Subnets

The following guidelines and limitations apply when defining L3Out subnets:

• Use the exact prefix match for Import Route Control Subnets and Export Route Control Subnets or use the 0.0.0.0/0 aggregate to match all routes.

• The same subnet should not be used for different external EPGs.

• When creating a subnet, the Export Route Control Subnets and Import Route Control Subnets allow Aggregate Export and Aggregate Import respectively with subnet configured to 0.0.0.0/0.

## Recommended Procedures for Defining L3Out Subnets

Layer 3 outside networks (L3Outs) for external EPGs are used to control which prefixes are allowed into or out of the fabric and which external networks are allowed to communicate with internal or other external networks.

You can configure the following L3Out subnet options:
• **Export Route Control Subnet**—Controls which external networks are advertised out of the fabric, using route-maps and IP prefix-lists.

• **External Subnets for the External EPG**—Sets the classifier for the external EPG. The rules and contracts assigned in this external EPG apply to networks matching this subnet.

• **Shared Route Control Subnet**—Controls which external prefixes are advertised to other tenants for shared services.

• **Shared Security Import Subnet**—Sets the classifier for the subnets in the VRF where the routes are advertised.

**Export Route Control**

---

**Note**

This section refers to the Cisco APIC GUI at [Tenants > tenant-name > Networking > External Routed Networks > Create Routed Outside > External EPG Networks > Create External Network > Subnet > Create Subnet > Export Route Control Subnet.](#)

Export route control determines which transit prefixes are advertised on the Layer 3 outside network associated with an external EPG. An IP prefix-list is created on the border leaf for each subnet that is defined here. A route-map is configured with all IP prefix-lists and is used for redistribution into OSPF or EIGRP L3Outs, or as an outbound route-map for BGP L3Outs.

The following command output shows the route-maps created:

```
BL-1# show ip ospf vrf T1:ctx1
Routing Process default with ID 1.1.1.103 VRF T1:ctx1
Stateful High Availabilty enabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
Table-map using route-map exp-ctx-2883588-deny-external-tag
Redistributing External Routes from
    static route-map exp-ctx-st-2883588
direct route-map exp-ctx-st-2883588
    bgp route-map exp-ctx-proto-2883588
eigrp route-map exp-ctx-proto-2883588
```

If no subnets are added to export route control, a route-map is not created. In the following example, no routes are redistributed into OSPF because the route-map being referenced by the redistribution command does not exist.

```
BL-1# show route-map exp-ctx-st-2883588
% Policy exp-ctx-st-2883588 not found
```

The route-map and IP prefix-list are created when the first subnet is added to export route control.

For example, if you set the Create Subnet dialog box to use the 172.16.25.0/24 IP address and the scope set to Export Route Control Subnet, the following route-map and IP prefix-list are displayed in the output of the show route-map command:

```
BL-1# show route-map exp-ctx-proto-2883588
route-map exp-ctx-proto-2883588, permit, sequence 7801
    Match clauses:
        ip-address prefix-lists: IPv6-deny-all
        IPv4=proto16390-2883588=exc-xt6-inferred-export-dst
    Set clauses:
        tag 4294967295
```
BGP L3Outs do not use redistribution to advertise the transit routes because routes received from L3Outs are already redistributed into MP-BGP. Therefore, they already exist in the BGP table on the border leaf. BGP uses outbound route-maps for export route control. The same rules apply to creation of the route-map and IP prefix-list. They are not created until the first export route-control subnet is configured. The following example shows the resulting outbound route-map:

```
Inbound route-map configured is permit-all, handle obtained
Outbound route-map configured is exp-l3out-BGP2-peer-2293764, handle obtained
```

When configuring export route-control subnets you must specify the exact prefix match. For example, an export route-control subnet of 172.16.0.0/16 only matches route 172.16.0.0/16. It does not match longer prefix length routes, such as 172.16.1.0/24 or 172.16.2.0/24. An exception to this is the 0.0.0.0/0 subnet. If you use this subnet, you can enable **Aggregate Export** on the *Create Subnet* dialog box. When aggregate export is enabled, the route control subnet matches all routes. If aggregate export is not enabled with the 0.0.0.0/0 subnet, then only the default route is advertised.

---

**Note**

Export route control is not used to advertise tenant subnets. Instead you configure that in the bridge domain/EPG subnet policy. The *Advertised Externally* option is used to advertise tenant subnets externally on the L3Out. See the *Create Subnet* dialog box at Tenants > tenant-name > Networking > Bridge Domains > BD-name > Create Subnet.

For example, on this *Create Subnet* dialog box, if you configure the *Gateway IP* address 10.1.1.1/24 and enable the *Advertised Externally* option, the system adds the tenant subnet to a static-redistribution route-map.

---

**Import Route Control**

Import route control subnets control the prefixes that are allowed into the fabric. By default **Import Route Control Subnets** are disabled by default and all prefixes are allowed into the fabric. When import route control is enabled, it is only used for BGP and OSPF L3Outs. EIGRP L3Outs do not use import route control. Routes learned from that protocol are always allowed into the fabric. Import route control uses an inbound route-map configured for the BGP neighbor or the OSPF area. If import route control is not enabled, the route-map permits traffic from all prefixes.

```
Inbound route-map configured is permit-all, handle obtained
Outbound route-map configured is exp-l3out-BGP2-peer-2293764, handle obtained
```

---

**Note**

You enable import route control when you create an L3Out, at Tenants > tenant-name > Networking > External Routed Networks > Create Routed Outside.

For example, in the *Create Routed Outside* dialog box, if you enable BGP or OSPF and enable **Route Control Enforcement** for Import, an inbound route-map is configured for the BGP neighbor. Similar to export route control, the route-map is not created until an import route-control subnet is added to the L3Out. Import route control follows the same rules as export route control (allowing exact prefix match or an aggregate for the 0.0.0.0/0 subnet).

In the following example, similar route-control subnets have been configured for the inbound and outbound route-maps:
Inbound route-map configured is imp-l3out-BGP2-peer-2293764, handle obtained
Outbound route-map configured is exp-l3out-BGP2-peer-2293764, handle obtained

Import route control is not only used to filter routes. It is also used to apply route-map match and set statements to route-maps. Use Set Rules for Route Maps and Match Rules for Route Maps to create set and match statements for route-maps. The Cisco APIC then assigns the profile to a route control profile.

In the following example, (at Tenants > tenant-name > Networking > External Routed Networks > Set Rules for a Route Map) you create a rule to set the BGP local preference value and assign this to the default import route control profile.

The default import route-control profile only applies the inbound route-map if import route control is enabled.

For example, if you set import route control for the 0.0.0.0/0 aggregate subnet, this matches all prefixes and permits them into the fabric. It also sets the BGP local preference to 200. See the following show command output:

```
BL-1# show route-map imp-l3out-BGP2-peer-2293764
route-map imp-l3out-BGP2-peer-2293764, permit, sequence 8001
  Match clauses:
    ip address prefix-lists: IPv6-deny-all
    IPv4-peer49153-2293764-agg-ext-in-default-import4rct10pfx-only-dst
  Set clauses:
    local-preference 200
```

You can also apply set rules for specific prefixes while still allowing all other prefixes into the fabric. In this case, (at Tenants > tenant-name > Networking > External Routed Networks > Create Routed Outside > External EPG Networks > Create Route Profile) create a different route control policy instead of using the default import policy. (Select Match Prefix and Routing Policy and set the order to 0.)

To apply this policy to specific prefixes, first create an import route control policy for the 0.0.0.0/0 aggregate subnet to match all prefixes, and use an empty default import route control profile. Then, configure an import route control policy for the prefixes that will use the route-control profile to set the BGP local preference. For example, enter the subnet, 10.206.19.0/24, and in the Route Control Profile field, identify the route control profile you just created for exceptions.

The route-map is created in the correct order to set the local preference for the specific route and match all other routes in the last sequence, as displayed in the following example showing the route-map creation sequence:

```
BL-1# show route-map imp-l3out-BGP2-peer-2293764
route-map imp-l3out-BGP2-peer-2293764, permit, sequence 2001
  Match clauses:
    ip address prefix-lists: IPv6-deny-all
    IPv4-peer49153-2293764-exc-ext-in-local-pref-3001local-pref-3000pfx-only dst
  Set clauses:
    local-preference 300
route-map imp-l3out-BGP2-peer-2293764, permit, sequence 8001
  Match clauses:
    ip address prefix-lists: IPv6-deny-all
    IPv4-peer49153-2293764-agg-ext-in-default-import4all-routes0pfx-only-dst
  Set clauses:

BL-1# show ip prefix-list
IPv4-peer49153-2293764-exc-ext-in-local-pref-3001local-pref-3000pfx-only dst
  ip prefix-list IPv4-peer49153-2293764-exc-ext-in-local-pref-3001local-pref-3000pfx-only dst: 1 entries
    seq 2 permit 10.206.19.0/24

BL-1# show ip prefix-list
IPv4-peer49153-2293764-agg-ext-in-default-import4all-routes0pfx-only-dst
  ip prefix-list IPv4-peer49153-2293764-agg-ext-in-default-import4all-routes0pfx-only dst: 1
```
External Subnets for an External EPG

This section refers to the Create Subnet dialog box at Tenants > tenant-name > Networking > External Routed Networks > Create Routed Outside > External EPG Networks > Create External Network > Create Subnet > External Subnets for the External EPG.

The external subnets for an external EPG are used to define the subnets that should be classified to the external EPG. This policy does not affect routing. It is similar to an Access Control List (ACL) that assigns a prefix to the class id (pcTag) of the external EPG.

Even though the external subnet for the external EPG is configured with the L3Out, the ACL is applied at the VRF level. This means that if a prefix is configured for L3Out-1 and traffic with a source address matching that prefix arrives on L3Out-2 the traffic is classified to the external EPG of L3Out-1. The following diagram explains this behavior:

*Figure 70: Action of External EPG ACL*

In this example, two Layer 3 outside networks are both using the 0.0.0.0/0 subnet. Traffic arriving on L3Out-2 is classified to the external EPG of L3Out-1 and is permitted to access the Web EPG even though there is no contract configured for the external EPG of L3Out-2.
If networks from L3Out-2 should not access the web EPG, then specific prefixes should be configured to match the subnets expected on each L3Out. The following example shows specific subnets configured for each L3Out:

*Figure 71: Specific Subnets Defined for Each L3Out*

![Figure 71: Specific Subnets Defined for Each L3Out](image)

External subnets for an external EPG are longest prefix-match subnets. This allows you to configure multiple external EPGs under one L3Out and apply different security policies (contracts) to each external EPG. The following table shows three external EPGs configured under the same L3Out. EPG-2 and EPG-3 are configured with subnets that are longer prefix-match subnets in the same subnet range as EPG-1.

*Table 16: Three External EPGs under the Same L3Out*

<table>
<thead>
<tr>
<th>L3Out</th>
<th>External Subnet for the External EPG</th>
<th>Contract</th>
</tr>
</thead>
<tbody>
<tr>
<td>External EPG-1</td>
<td>192.168.0.0/16</td>
<td>Contract-1</td>
</tr>
<tr>
<td>External EPG-2</td>
<td>192.168.1.0/24</td>
<td>Contract-2</td>
</tr>
<tr>
<td>External EPG-3</td>
<td>192.168.1.1/32</td>
<td>Contract-3</td>
</tr>
</tbody>
</table>
Shared Route-Control Subnets

This section refers to the Create Subnet dialog box at Tenants > tenant-name > Networking > External Routed Networks > Create Routed Outside > External EPG Networks > Create External Network > Create Subnet > Shared Route Control Subnet.

Shared route-control subnets are used with shared L3Outs. They control which external prefixes are advertised to other VRFs, which have a contract interface to the shared L3Out. This subnet type is similar to export route control with one exception: the Aggregate Shared Routes option applies to any subnet, not just the 0.0.0.0/0 subnet. For example, if you configure subnet 192.168.0.0/16 with the aggregate shared routes option, this matches the 192.168.0.0/16 subnet and all 192.168.0.0 subnets with longer prefix lengths. This is equivalent to configuring an IP prefix-list with the le 32 keyword (less than or equal to).

Shared Security-Import Subnets

This section refers to the Create Subnet dialog box at Tenants > tenant-name > Networking > External Routed Networks > Create Routed Outside > External EPG Networks > Create External Network > Create Subnet > Shared Security Import Subnet.

Shared security-import subnets are used with shared L3Out configuration, not used for routing control. This setting configures an ACL similar to Export Route-Control Subnets, but the ACL is configured in the VRF that is consuming the shared L3Out. This is a longest prefix-match subnet.

Verifying L3Out Subnet Definitions

The following commands can be used to verify configuration of the L3Out subnets on the border leaf:

- `show ip ospf vrf vrf-name`
- `show ip eigrp vrf vrf-name`
- `show ip bgp neighbor vrf vrf-name`
- `show route-map [route-map-name]`
- `show ip prefix-list [ip-prefix-name]`

Additional References for Defining L3Out Subnets

For more information on L3Out definitions, see the Cisco Application Centric Infrastructure Fundamentals Guide at the following URL:

Cisco AVS Distributed Firewall

About Cisco AVS Distributed Firewall

The Distributed Firewall is part of Cisco Application Virtual Switch (AVS) at the hypervisor kernel level and works in conjunction with the Cisco Application Centric Infrastructure (ACI) hardware for policy enforcement. The Distributed Firewall keeps track of the state of the network connection, traversing across it, and distinguishes legitimate packets for different types of TCP connection.

The Cisco AVS Distributed Firewall is completely assisted by Cisco ACI hardware. This combination of hardware and software provides greater performance and agility. In the Cisco ACI solution, leaf switches act as a policy store, which does not incur any performance penalty as the policies are processed in the hardware.

Distributed Firewall Behavior

In a Cisco ACI fabric, contracts using subjects and filters between consumer and provider EPGs are used to allow traffic. For example, the administrator creates a policy to allow traffic from any source port to destination port 80. As soon as the policy is configured in the Cisco Application Policy Infrastructure Controller (APIC), a reflexive ACL (access control list) entry from the provider to the consumer is automatically programmed in the Cisco ACI hardware. This reflexive ACL entry is necessary to allow the reverse traffic to flow. However, the leaf switch allows the provider (source port 80) to connect to any client destination port, which might not be desirable for some data centers. That is because endpoints in a provider EPG might initiate a SYN attack or a port-scan to the endpoints in the consumer EPGs using its source port 80.
The Distributed Firewall, with the help of the physical leaf switches, will not allow such SYN attacks. The leaf switch evaluates the packet and allows TCP packets only if the ACK flag is set, which prevents SYN attacks. Cisco AVS maintains the connection table to track the flow and allows TCP packets only if Cisco AVS has flow entry.

TCP Packets from Provider to Consumer

You can enable the Distributed Firewall feature on both the hardware leaf switch and Cisco AVS to prevent SYN attacks and SYN and ACK attacks from the provider.

The following figure illustrates how to prevent a SYN attack from the provider:
Figure 74: Preventing SYN attack from the Provider

In this case, the following behavior occurs:

- If the SYN packets do not have the ACK bit set, then the hardware leaf switch drops the packets.
- If the SYN and ACK packets have the ACK bit set, then the hardware leaf permits the packets, but flow entry does not exist on AVS at the provider side. Therefore, Cisco AVS drops the packets.

The following figure illustrates how to prevent a SYN and ACK attack from the provider:

Figure 75: Preventing a SYN and ACK Attack from the Provider

In this case, the following behavior occurs:

- If the data packets have the ACK bit set, then the hardware leaf switch permits the packets. If the connection is established, a flow entry exists on AVS. Therefore, the packets are permitted.
- If the RST packets have the ACK bit set, then they are handled similar to the data packets.
- If the FIN packets have the ACK bit set, then they are handled similar to the data packets. The FIN packets without the ACK bit set will be dropped by the hardware leaf switch.
The handling of FIN packets without the ACK bit set differs based on the type of the operating system, which enables such packets to be used for a FIN scan attack to determine the operating system. Dropping such packets can prevent this attack.

**Seamless FTP Traffic Handling**

The Distributed Firewall provides a stateful inspection capability for the FTP protocol. The Distributed Firewall snoops the FTP control connection (server TCP port 21) to get the data connection details (client IP and client port) and allows the FTP data connection (server TCP port 20) only for that flow. The Distributed Firewall supports only active FTP mode handling. No special handling is done for the passive FTP mode.

The following figure illustrates seamless FTP traffic handling:

*Figure 76: Seamless FTP Traffic Handling*

---

**Guidelines and Limitations for Cisco AVS Distributed Firewall**

The following guidelines and limitations apply when using Cisco AVS Distributed Firewalls:

- Reflective ACL in the hardware is programmed to allow TCP packets only if the ACK flag is set.
- In receiving the first TCP SYN packet, Cisco AVS creates a flow table entry. The Cisco AVS drops packets if it does not have flow entry.
- Cisco AVS maintains the connection table to track the flow. Cisco AVS allows TCP packets only if it has flow entries.
- We recommend that you use vmxnet3 adapters for the VMs when using Distributed Firewall. We also recommend that you use vmxnet3 adapters in scale setups to increase the DVSLargeHeap size to its maximum (64 on 5.1 hosts and 128 on 5.5 hosts). You need to reboot the host for the change to take effect. For more information about using vmxnet3 adapters for scale setups, see the following VMware knowledge base article:

  *Error message is displayed when a large number of dvPorts are in use in VMware ESXi 5.1.x (2034073).*

**Configuration Examples for Cisco AVS Distributed Firewall Using the GUI**

You configure the Distributed Firewall by choosing one of the following modes:

- Enabled—Enforces the Distributed Firewall.
• Disabled—Does not enforce the Distributed Firewall. Use this mode only if you do not want to use the Distributed Firewall. Disabling the Distributed Firewall removes all flow information on the Cisco AVS.

• Learning—Cisco AVS monitors all TCP communication and creates flows in a flow table, but does not enforce the firewall. Learning is the default firewall mode in Cisco AVS Release 5.2(1)SV3(1.5) and Release 5.2(1)SV3(1.10). Learning mode provides a way to enable the firewall without losing traffic.

The following procedure provides an example of configuring the Cisco AVS Distributed Firewall with the Enabled mode using the advanced GUI mode.

Procedure

**Step 1**
Reflective ACL in the hardware is programmed to allow TCP packets only if the ACK flag is set. The following steps demonstrate how to configure a leaf switch to check the ACK flag:

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

b) In the **Navigation** pane, expand **tenant_name > Security Policies > Filters**.

   The **Security Policies - Filters** panel appears in the **Work** pane. Your filters are displayed as rows inside a summary table.

c) Click the table row to display the **Filter** panel.

   The **Entries** table is displayed at the bottom of the **Filter** panel with a list of network traffic classification properties. To configure a leaf switch to check the ACK flag and allow TCP packets, the **Stateful** check box in the **Entries** table must be checked (set to True). By default, the **Stateful** check box is unchecked (set to False).

d) To check the **Stateful** check box, double-click on the row in the **Entries** table that represents the filter you want to configure. The filter will have tcp in the **Protocol** column and False in the **Stateful** column.

   The chosen row expands and enables you to edit the network traffic classification properties.

e) Put a check in the **Stateful** check box.

f) Click **Update**.

**Step 2**
In receiving the first TCP SYN packet, Cisco AVS creates a flow table entry. If Cisco AVS does not have a flow entry, it drops the packets. The following steps demonstrate how to configure Cisco AVS to enable the distributed firewall and maintain a connection table to track the flow:

a) On the menu bar, choose **Fabric > Access Policies**.

b) In the **Navigation** pane, choose **Interface Policies > Policies > Firewall > default**.

   The **Firewall Policy - default** panel appears.

c) In the **Mode** field, click the **Enabled** button. This property is referred to by VMM domain vSwitch policies. By default the **Mode** is **Learning**.

d) From the menu bar, choose **VM NETWORKING > Inventory > VMware > ACI_AVS_name**.

e) From the **ACI_AVS_name** pane, in the **VSwitch Policy** section, ensure the **Firewall Policy** field is **default**.

   If the **Firewall Policy** field is not set to **default**, you must be in the advanced GUI mode to change it.
TCP Packet Handling Example

The example below demonstrates how TCP packets are handled if the distributed firewall feature is enabled on both the leaf switches and the Cisco AVS (also see Configuration Examples for Cisco AVS Distributed Firewall Using the GUI, on page 178).

If the SYN packets do not have the ACK bit set, the leaf switch drops the packets. If the SYN and ACK packets have the ACK bit set, the leaf switch permits the packets, but the flow entry does not exist on Cisco AVS at the provider side. This causes the Cisco AVS to drop the packets.

*Figure 77: Prevent SYN attack from Provider*

If the data packets have the ACK bit set, the leaf switch permits the packets. If the connection is established, a flow entry exists on Cisco AVS and the packets are permitted. If the RST packets also have the ACK bit set, they are handled similarly to the data packets.

FIN packets with the ACK bit set are also handled similarly to the data packets. The FIN packets without the ACK bit set are dropped by the leaf switch.

*Figure 78: Prevent SYN and ACK attack from Provider*
• The handling of FIN packets without the ACK bit set differs based on the type of the operating system. So it can be used for FIN scan attacks to determine the operating system.

• Dropping FIN packets without the ACK bit set can prevent such an attack.

**FTP Traffic Handling Example**

Distributed firewall provides a stateful inspection capability for FTP. Distributed firewall snoops the FTP control connection (Server TCP port 21) to get the data connection details (client IP and client port) and to allow the FTP data connection (Server TCP port 20) only for that flow. Support is only for active-FTP mode handling. No special handling will be done for passive-FTP mode.

*Figure 79: Seamless FTP Traffic Handling*

**Additional References for Cisco AVS Distributed Firewall**

For more information on AVS Distributed Firewalls, see the *Cisco ACI Virtualization Guide* at the following URL:

Additional References for Cisco AVS Distributed Firewall
CHAPTER 11

Miscellaneous Implementation

• The Basic GUI and the Advanced GUI, on page 183
• Migrating Existing Networks to Cisco ACI, on page 184

The Basic GUI and the Advanced GUI

About the Basic GUI and the Advanced GUI

The Advanced Mode is the same GUI that has existed since 1.0 code. It represents a 1:1 mapping with the underlying object model. As of Cisco Policy Infrastructure Controller Release 1.2(1), there is now an option to utilize a Basic Mode. The Basic Mode intends to mask some of the complexity associated with Cisco Application Centric Infrastructure (ACI) constructs over the course of configuration. By doing so, the Basic Mode brings a set of limitations in what can and cannot be accomplished for configuration.

The main differences between the Advanced Mode and the Basic Mode are in the workflows that need to be performed to achieve the same configuration. For example, with the Basic Mode, you configure one port at a time, which means the GUI creates one object for each port. The Advanced Mode can be used to create multiple relationships with existing objects, where applicable, and do wholesale configurations using policies and profiles.

Prerequisites for the Basic GUI vs the Advanced GUI

This section contains the prerequisites for each Cisco APIC GUI mode:

• The Basic Mode is available on Cisco APIC Release 1.2(1) and later.
• The Advanced Mode is the same GUI that has been available since product launch.

Guidelines and Limitations for Basic GUI vs Advanced GUI

This section contains the guidelines and limitations for using the Cisco APIC GUI modes:

• If a Cisco ACI fabric was initially deployed on the Advanced Mode, you should continue to use the Advanced Mode for configuration deployment.
• If a Cisco ACI fabric was deployed with the Basic Mode, you should continue to use the Basic Mode configuration deployment.

• Switching between the Basic Mode and Advanced Mode configurations within the same fabric is not supported. Going back and forth between GUI modes while performing configurations can cause undesired relationships between objects if great care is not taken.

• The Basic Mode is designed for usage on small scale, greenfield deployments. This is due to the fact that every instance of policy created within the Basic Mode is a new instance. The Basic Mode is not built around policy reuse.

• L4-L7 services configuration is not available within the Basic Mode.

• Objects created due to the Basic Mode will show up with a prefix of “__ui__” when viewed from the Advanced GUI. They cannot be removed in the Advanced GUI. For the steps to remove unwanted __ui__ objects, see Troubleshooting Unwanted __ui__ Objects in the Cisco APIC Troubleshooting Guide.

• The Basic Mode and the NX-OS-Style CLI utilize the same set of scripts to perform configuration. As such, the NX-OS-Style CLI has the same limitations associated with the Basic Mode.

Verifying the Basic GUI vs the Advance GUI

The current Cisco APIC GUI mode is specified in the top-right corner of the APIC GUI when logged in.

Additional References for Using the Basic GUI and Advanced GUI

• For Basic Mode and Advanced Mode configuration examples, see the Cisco APIC Getting Started Guide at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html

Migrating Existing Networks to Cisco ACI

About Migrating Existing Networks to Cisco ACI

The network centric migration process consists in interconnecting the existing network (built based on STP, vPC, or FabricPath technologies) to a newly developed Cisco Application Centric Infrastructure (ACI) POD with the end goal of migrating applications or workloads between those environments.

Prerequisites for Migrating Existing Networks to Cisco ACI

To accomplish an application migration task, it is required that you map traditional networking concepts (VLANs, IP subnets, VRFs, etc.) to new Cisco ACI constructs such as endpoint groups (EPGs), bridge domains (BDs), and private networks.
Recommended Configuration Procedure for Migrating Existing Networks to Cisco ACI

The steps of the Cisco ACI network-centric migration methodology are described as follows:

**Procedure**

**Step 1**  
Design and deploy the new Cisco ACI POD; it is likely that the size of such a deployment is initially small with plans to grow in time with the number of applications that are migrated.  
A typical Cisco ACI POD consists of at least two spine switches and two leaf switches and is managed by a cluster of Cisco APIC controllers.

**Step 2**  
Perform the integration between the existing DC network infrastructure and the new Cisco ACI POD.  
Layer 2 and Layer 3 connectivity between the two networks is required to allow successful applications and workload migration across the two network infrastructures.

**Step 3**  
Migrate the workloads between the existing network and the new network.  
It is likely that this application migration process may take several months to complete (depending also on the number and complexity of the applications being migrated), so communication between new and existing networks through the Layer 2 and Layer 3 connections previously mentioned is utilized during this phase.

**Additional References for Migrating Existing Networks to Cisco ACI**

For more information, see the *Migrating Existing Networks to Cisco ACI* and the *FabricPath to ACI Migration Cisco Validated Design Guide* at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html
PART III

Operations

- ACI Constructs Operations, on page 189
- Layer 4 to Layer 7 Operations, on page 195
- Miscellaneous Operations, on page 199
AAA RBAC and Roles

About AAA RBAC and Roles

The Application Policy Infrastructure Controller (APIC) provides access according to a user's role through role-based access control (RBAC). A Cisco Application Centric Infrastructure (ACI) fabric user is associated with the following role components:

- A set of roles
- For each role, a privilege type: no access, read-only, or read-write
- One or more security domain tags that identify the portions of the management information tree (MIT) that a user can access

The ACI fabric manages access privileges at the managed object (MO) level. A privilege is an MO that enables or restricts access to a particular function within the system. For example, fabric-equipment is a privilege bit. This bit is set by the APIC on all objects that correspond to equipment in the physical fabric.

A role is a collection of privilege bits. For example, because an "admin" role is configured with privilege bits for "fabric-equipment" and "tenant-security," the "admin" role has access to all objects that correspond to equipment of the fabric and tenant security.

A security domain is a tag that is associated with a certain subtree in the ACI MIT object hierarchy. For example, the default tenant "common" has a domain tag "common." Similarly, a special domain tag "all" includes the entire MIT object tree. An admin user can assign custom domain tags to the MIT object hierarchy. For example, a "solar" domain tag is assigned to the tenant solar. Within the MIT, only certain objects can be tagged as security domains. For example, a tenant can be tagged as a security domain, but objects within a tenant cannot.

If a virtual machine management (VMM) domain is tagged as a security domain, the users contained in the security domain can access the correspondingly tagged VMM domain. For example, if a tenant named "solar" is tagged with the security domain called "sun" and a VMM domain is also tagged with the security domain called "sun," then users in the solar tenant can access the VMM domain according to their access rights.
Prerequisites for AAA RBAC and Roles

You must meet the following prerequisites to use AAA role-based access control (RBAC) and roles:

- Find the API documentation at the following URL:
  

- Deploy an authentication domain (LDAP, RADIUS, TACACS+) that is reachable by out-of-band or in-band management from the Application Policy Infrastructure Controller (APIC).

Guidelines and Limitations for AAA RBAC and Roles

The following guidelines and limitations apply for AAA role-based access control (RBAC) and roles:

- If you change the default authentication domain, then you must specify any domain other than the default when logging in to the API, GUI, or CLI.

  For the API, the syntax is as follows:
  
  `apic:domain\your_username`

  For the CLI, the syntax is as follows:
  
  `apic#domain\your_username`

- You should leave the "fallback" domain as local authentication in case an issue arises with the remote authentication server. If that is done, you can specify the local domain by using the above syntax, but with the domain specified as "fallback." For example:

  `apic# fallback\your_local_username`

- The APIC Management Information Model Reference lists every privilege that has read and write access to a given class. For example, looking at the class of a bridge domain (fvBD), you get the following information:

  Class fv:BD (CONCRETE)

  Class ID:1887
  Class Label: Bridge Domain
  Encrypted: false - Exportable: true - Persistent: true - Configurable: true
  Write Access: [admin, tenant-connectivity-l2]
  Read Access: [admin, nw-svc-device, nw-svc-policy, tenant-connectivity-l2,
  tenant-connectivity-mgmt, tenant-epg, tenant-ext-connectivity-l2,
  tenant-network-profile,
  tenant-protocol-l2, tenant-protocol-l3]
  Creatable/Deletable: yes (see Container Mos for details)
  Semantic Scope: EPG
  Semantic Scope Evaluation Rule: Explicit
  Monitoring Policy Source: Explicit
  Monitoring Flags : [ IsObservable: true, HasStats: true, HasFaults: true, HasHealth: true,
  HasEventRules: false ]

  The information indicates that for a user to be able to write changes to a bridge domain, the user must have a role that contains either the "admin" bits or the "tenant-connectivity-l2" bits. These privileges can be found when viewing pre-existing roles or creating new ones.

  - Security domains allow a user to be exposed to only specific branches of the Management Information Tree (MIT). Typically, this allows ACI administrators to expose only specific tenants to users to
give the fabric the aspect of multi-tenancy in that users only have access to view and make changes to their own tenant.

- A fabric-wide administrator uses RBAC rules to selectively expose physical resources to users that otherwise are inaccessible because they are in a different security domain. While an RBAC rule exposes an object to a user in a different part of the management information tree, it is not possible to use the CLI to navigate to such an object by traversing the structure of the tree. However, as long as the user knows the distinguished name of the object that is included in the RBAC rule, the user can use the CLI to locate the object by using the `MO find` command.

- Modifying the "all" security domain to give a user access to resources outside of that user's security domain is bad practice. Such a user will then have access to resources that are provisioned for other users.

### Recommended Configuration Procedure for AAA RBAC and Roles

The following information applies when configuring AAA role-based access control (RBAC) and roles:

- Security domains can be tied to exactly one tenant for multi-tenancy.

- When utilizing remote authentication domains, a string attribute is needed to tie the user account to the security domain. The attribute is typically referred to as the CiscoAVPair, but can be named anything as long as the attribute is set to a type of "Case Sensitive String." This configuration is done on the authentication server itself, not on the Cisco Application Centric Infrastructure (ACI) fabric.

### Verifying the AAA RBAC and Roles Using the GUI

The following procedure verifies the assigned AAA roles using the Application Policy Infrastructure Controller (APIC) GUI.

**Procedure**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>On the menu bar, choose <code>welcome, user_name &gt; AAA &gt; View My Permissions</code>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the <code>User Permissions</code> dialog box, you can view any security domains to which you have access, along with the tenants that are associated specifically to those domains.</td>
</tr>
</tbody>
</table>

### Configuration Examples for AAA RBAC and Roles Using the GUI

The following procedure provides an example of configuring AAA role-based access control (RBAC) and roles using the Application Policy Infrastructure Controller (APIC) GUI.

**Procedure**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Create a security domain. On the menu bar, choose <code>Admin &gt; AAA</code>.</th>
</tr>
</thead>
</table>
Step 3  In the Work pane, choose Action > Create Security Domain.
Step 4  In the Create Security Domain dialog box, fill out the fields as necessary.
Step 5  Associate the security domain with a tenant. On the menu bar, choose Tenants > All Tenants.
Step 6  In the Work pane, double-click the tenant's name.
Step 7  In the Security Domains section, put a check in the check boxes that correspond to the security domain that you want to associate with the tenant.
Step 8  Create the RBAC rules. On the menu bar, choose Admin > AAA.
Step 9  In the Navigation pane, choose Security Management > RBAC Rules.
Step 10 In the Work pane, choose Action > Create RBAC Rule.
Step 11 In the Create RBAC Rule dialog box, fill out the fields as necessary. You must specify the distinguished name (DN) of the object to be acted upon and the domain to add the rule. You can also specify write privileges for this RBAC rule.

### Additional References for AAA RBAC and Roles

For more information about AAA within Cisco Application Centric Infrastructure (ACI), see the *Cisco Application Centric Infrastructure Fundamentals Guide*.

For more information about configuring authentication domains in ACI see the *Configuring TACACS+, RADIUS, and LDAP for Cisco APIC Access* knowledge base article.

You can find the specified documentation at the following URL:


### Endpoint Loop Protection

#### About Endpoint Loop Protection

The endpoint loop protection feature enables you to specify the number of times an endpoint can move before taking one of the two following actions:

- Disable endpoint learning within the bridge domain.
- Disable the port that the endpoint is connected to.

The recommendation is to enable endpoint loop protection using the following default parameters:

- Loop detection interval: 60
- Loop detection multiplication factor: 4
- Action: Port Disable

The above parameters state that if an endpoint moves more than four times within a sixty second period, then the endpoint loop protection will take the specified action of disabling the port.
Configuration Example for Endpoint Loop Protection

The following procedure provides an example of configuring endpoint loop protection using the Application Policy Infrastructure Controller (APIC) GUI.

**Procedure**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>On the menu bar, choose Fabric &gt; Access Policies.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the Navigation pane, choose Global Policies &gt; EP Loop Protection Policy.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the Work pane, choose Enable and enter the appropriate values in each field of the EP Loop Protection Policy panel.</td>
</tr>
<tr>
<td>Step 4</td>
<td>To bring a disabled port back up after a specified time, configure automatic error disable recovery. In the Navigation pane, choose Global Policies &gt; Error Disabled Recovery Policy.</td>
</tr>
<tr>
<td>Step 5</td>
<td>In the Work pane, double-click Frequent EP Move.</td>
</tr>
<tr>
<td>Step 6</td>
<td>Put a check in the Frequent EP Move check box.</td>
</tr>
<tr>
<td>Step 7</td>
<td>Click Update.</td>
</tr>
</tbody>
</table>
CHAPTER 13

Layer 4 to Layer 7 Operations

• Device Packages, on page 195

Device Packages

About Device Packages

A device package is used to insert and configure network service functions on a network service appliance (device). A device package contains the following components:

• Device Specification (XML)—The configuration of the Application Policy Infrastructure Controller (APIC) is represented as an object model consisting of a large number of managed objects (MOs). A device type is defined by a tree of MOs with a meta device (MDev) at the root.

• Device Script (py)—The integration between the APIC and a device is performed by a device script, which maps APIC event function calls that are defined in the device script.

When you upload a device package to the APIC, the APIC creates a hierarchy of MOs that represent the device and validates the device script interface.

Guidelines and Limitations for Device Packages

The following guidelines and limitations apply for device packages:

• Device packages are managed by third party vendors.

• If the major version (the naming property of class vnsMDev) changes, uploading the new device package will create a new package. For example, if the original Cisco ASA package distinguished name was "uni/infra/mDev-CISCO-ASA-1.0" and the new package version changed to "2.0", then the new distinguished name will be "uni/infra/mDev-CISCO-ASA-2.0".

• When importing a device package with a major version change, old service graphs and old device clusters will continue to point to the old package and will continue working. New service graphs and new device clusters can choose to use either the old or new device package. However, switching an old service graph or device cluster to the new package will be disruptive.

• Changing the minor version (a property called minorversion in the MO called DevScript) does not change the distinguished name of the device package or the vnsMDev class.
• Uploading a new device package with a **minor version** change overwrites the existing device package. All graphs and device clusters pointing to the old device package start pointing to the new package automatically. The upgrade is non-disruptive and there should be no impact for existing service graphs or device clusters.

• A **minor version** change is the default recommendation for partners for any new device package revisions.

• When using a device package, a device cluster can be managed by only one device package at any time.

• A node in a service graph can associate to only one device package at any time.

• The node in the service graph and the associated device cluster should point to the same device package. That is, you cannot have a node in a service graph that points to the old device package while the device cluster points to the new package.

• The Application Policy Infrastructure Controller (APIC) treats the version field as an opaque string. A change from "1.0" to "2.0" as well as "1.0" to "1.1" both look the same to the APIC and are considered to be a major version change.

• The APIC images are backward compatible with old device packages.

• If a device package is already uploaded and the APIC is upgraded, the old device package will continue to work without any disruption.

• Newer device packages might not work on older versions. In such cases, the device package upload step will fail with an appropriate error.

• Make sure that the device package is supported on the vendor device (hardware and software) and that the device package is compatible with the Cisco Application Centric Infrastructure (ACI) platform. For more information, see the **L4-L7 Compatibility List Solution Overview** document at the following location: [http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/solution-overview-c22-734587.html](http://www.cisco.com/c/en/us/solutions/collateral/data-center-virtualization/application-centric-infrastructure/solution-overview-c22-734587.html)

• Understand the differences between major/minor version changes on the device package and the impact when upgrading a device package.

• Understand the features to be configured through the APIC by way of the device package to the services appliance. For example, understand the features on the firewalls or load balancers that the administrator wishes to configure versus the features that are supported by the device package.

---

**Recommended Procedure for Importing a Device Package Using the GUI**

The following procedure imports a device package using the GUI. You must use the advanced GUI mode.

**Procedure**

**Step 1** On the menu bar, choose **L4-L7 Services > Packages**.

**Step 2** In the **Navigation** pane, choose **L4-L7 Service Device Types**.

**Step 3** In the **Work** pane, choose **Actions > Import Device Package**.

**Step 4** In the **Import Device Package** dialog box, click **Browse**.

**Step 5** In the **Open** dialog box, find and choose the device package that you want to import.
Step 6  Click **Open**. The Application Policy Infrastructure Controller (APIC) can take several seconds to open the device package.

Step 7  In the **Import Device Package** dialog box, click **Submit**.

The device package gets imported into the APIC. You can see the device package in the **Work** pane.

---

**Verifying the Device Package Versions**

The following procedure verifies a device package's versions using the GUI. You must use the advanced GUI mode.

**Procedure**

Step 1  On the menu bar, choose **L4-L7 Services > Packages**.

Step 2  In the **Navigation** pane, choose **L4-L7 Service Device Types > device_package_name**.

Step 3  In the **Work** pane, the **Major Version** field shows the major version of the device package. The **Minor Version** field shows the minor version of the device package.
Verifying the Device Package Versions
CHAPTER 14

Miscellaneous Operations

- API Inspector, on page 199
- Audit Logs, on page 200
- GUI Application Settings, on page 202
- Health Scores, on page 203
- Using the Cisco NX-OS Style CLI, on page 207
- Upgrading the Fabric, on page 208
- Snapshot and Configuration Rollback, on page 210
- Tags and Aliases, on page 211
- QuickStart in the Cisco APIC GUI, on page 213

API Inspector

About API Inspector

The API Inspector is a built-in tool in the Cisco Application Infrastructure Controller (APIC) GUI that allows you to capture internal REST API messaging as you perform tasks in the Cisco APIC GUI. The captured messages show the managed objects (MOs) being accessed and the JSON data exchanges of the REST API calls. You can use this data when designing Python API calls to perform similar functions.

Recommended Configuration Procedure for API Inspector

API Inspector is a debugging tool to help you understand the Application Policy Infrastructure Controller (APIC) GUI calls (GET or POST) and the managed object model. Based on the debug results, you can modify and repost the configuration using POSTMAN or create automated scripts or develop external applications that will use the API.

Verifying an API Inspector Configuration

The following is an example debug output displaying how Tenant Coca is created:

method: POST
url: https://10.10.10.1/api/node/mo/uni/tn-Coca.json
payload:{"fvTenant":{"attributes":{"dn":"uni/tn-Coca","name":"Coca","rn":"tn-Coca","status":"created"},"children":[]}}
response: {"totalCount":0,"imdata":[]}
Configuration Example for API Inspector

Use the following steps to access the API Inspector using the Advanced GUI or the Basic GUI:

**Procedure**

<table>
<thead>
<tr>
<th>Step 1</th>
<th>Log in to the Application Policy Infrastructure Controller (APIC) GUI.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the top-right corner of the GUI, click <strong>welcome user_name</strong> &gt; <strong>Show API Inspector</strong>. A new <strong>API Inspector</strong> screen displays.</td>
</tr>
</tbody>
</table>

Audit Logs

About Audit Logs

Within the Cisco Application Centric Infrastructure (ACI) fabric, the majority of what is viewable using the GUI is made possible through the underlying management information tree (MIT). Networking constructs and management constructs have been abstracted and represented as objects. The same applies to audit logs.

The audit logs within the ACI fabric are objects that are records of user-initiated events such as login, logout, object creation, and attribute changes under existing objects. These can be useful for tracking erroneous changes within the environment or simply for keeping an audit of changes that have occurred within the ACI fabric.

There is no configuration associated with audit logs.

Prerequisites for Audit Logs

You must meet the following prerequisites to use audit logs:

- Know that the class aaa:SessionLR represents fabric logouts and logins.
- Know that the class aaa:ModLR represents configuration changes.

Guidelines and Limitations for Audit Logs

The following guidelines and limitations apply for audit logs:

- Audit logs are enabled by default.
- Audit logs can be viewed within the GUI for specific objects.
- Audit logs can be extracted using the **moquery** command for fabric-wide change analysis.
Verifying the Audit Logs Using the GUI

The audit logs can be verified using the Advanced GUI or the Basic GUI for each object on which changes can be performed. The following procedure verifies changes to a specific tenant:

**Procedure**

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1</td>
<td>On the menu bar, choose Tenants &gt; All Tenants.</td>
</tr>
<tr>
<td>Step 2</td>
<td>In the Work pane, double-click the desired tenant's name.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the Navigation pane, choose Tenant tenant_name.</td>
</tr>
<tr>
<td>Step 4</td>
<td>In the Work pane, choose the History &gt; Audit Log tab. In the Audit Log pane, a list of all changes within this specific tenant is displayed.</td>
</tr>
<tr>
<td>Step 5</td>
<td>Double-click each item for more information, including the old and new states of the change.</td>
</tr>
</tbody>
</table>

Verifying Audit Logs Using the Object Model CLI

- When logged into the Application Policy Infrastructure Controller (APIC) object model CLI, you can extract a complete list of all configuration changes.
- The `moquery` command performs formatting against the text, so larger fabrics with many changes may cause this command to take a while to complete.

**Procedure**

Extract the configuration changes from audit logs.

**Example:**

```bash
apic1# moquery -c aaaModLR
```

The command output can be redirected to a file with the following syntax:

```bash
pod3-apic1# moquery -c aaaModLR > /tmp/audit_logs.txt
```

Additional References for Audit Logs

For additional information about audit logs, see the *Cisco Application Centric Infrastructure Fundamentals Guide* at the following URL:

GUI Application Settings

About the GUI Application Settings

The GUI application settings are a set of options designed to enhance the Application Policy Infrastructure Controller (APIC) GUI experience. These options are not relevant if configuration is being applied through some method other than the GUI.

Prerequisites for GUI Application Settings

You must meet the following prerequisites to use GUI application settings:

• The Cisco Application Centric Infrastructure (ACI) fabric must be built and accessible using HTTP or HTTPS.

Guidelines and Limitations for GUI Application Settings

The following guidelines and limitations apply for GUI application settings:

• Assuming you utilize the same browser for subsequent GUI access, GUI application settings are saved across sessions.

• For Remember Tree Selection, it only remembers the location if the last highlighted item within the GUI is an object. This can be viewed within the URL if there is an actual DN present.

Recommended Configuration Procedure for GUI Application Settings

The following GUI application settings are available for use:

• Remember Tree Selection—This setting enables the GUI to remember the last highlighted object, and display the full expanded tree view. Typically, every tab move causes all views to collapse. This setting allows the view to remain expanded on the last object viewed which is ideal for back and forth scenarios.

• Preserve Tree Divider Position—This setting is useful if the divider between the navigation pane and the working pane is altered. It retains the divider size across tab moves. Otherwise, the divider gets reset across tab moves.

• Disable Notification on Success—This setting is enabled by default. This prevents the dialog stating “changes saved successfully” from popping up after every configuration change.

• Disable Deployment Warning at Login—This setting is disabled by default. This controls the pop-up dialog box displayed upon every login indicating that deployment warning settings are disabled.

• Default Page Size for Tables—This setting enables you to set a global number for entries seen within each table. The default is 15, and it can be changed per table. Change it in this setting to set it for all tables being viewed.
Configuring GUI Application Settings

The following procedure configures the GUI application settings:

**Procedure**

**Step 1**
On the menu bar, on the far right, click `welcome user_name > Settings`.

**Step 2**
In the **Application Settings** dialog box, put a check in the check boxes for the desired settings.

**Step 3**
Click OK.
This completes the GUI application settings.

Verifying the GUI Application Settings

The GUI Application settings can be verified from the same location in which they were configured.

Health Scores

About Health Score

The Application Policy Infrastructure Controller (APIC) uses a policy model to combine the current status of all the manage objects including links, devices, and such into a health score. It provides the operator visibility and a quick overview into their entire Cisco Application Centric Infrastructure (ACI) system.

Cisco ACI fabric health information is available for the following areas of the system:

- **System**—Aggregation of system-wide health including pod health scores, tenant health scores, system fault counts by domain and type, and the Cisco APIC cluster health state.

- **Pod**—Aggregation of health scores for a pod (a group of spine and leaf switches) and pod-wide fault counts by domain and type.

- **Tenant**—Aggregation of health scores for a tenant, including performance data for objects such as applications and EPGs that are specific to a tenant and tenant-wide fault counts by domain and type.

- **Managed Object**—Health score policies for managed objects (MOs) which include their dependent and related MOs. These policies can be customized by an administrator.

The following figure displays a diagram describing the health scoring policy.
Prerequisites for Health Score

You must meet the following prerequisites to use the health score:

- Once the Cisco ACI fabric is operational, the system administrator or operators will be able to access the dashboard and to monitor the system by viewing the health score.

Guidelines and Limitations for Health Score

The following guidelines and limitations apply for health scores:
• Health scores are based on the faults generated in the fabric. Each fault reduces the health score based on the severity of the fault. The higher the fault severity is, the more penalty it will receive and impact the health score.

• Health score is calculated from a range of 0 to 100 (100 is the perfect health score).

• It is aggregated and available in different system level views.

• The health score of an application component can be distributed across multiple leaf switches. For example, a hardware fault impacts the health score of an application component.

• Starting with Cisco APIC release 1.2(2g), Cisco APIC supports the health score evaluation to ignore acknowledged faults, such as for those faults that can be safely ignored and prevent the health score from being degraded.

• You can modify the health score evaluation policy based on the penalty of the health score at the fault severity level. The health score evaluation policy can be configured as desired by navigating in the GUI to Fabric > Fabric Policies > Monitoring Policies > Common Policy > Health Score Evaluation Policies > Health Score Evaluation Policy_name. In the Work pane, under Properties, choose the desired settings.

**Recommended Configuration Procedure for Health Scores**

Cisco ACI health scores provide a quick check whether an issue being reported is confirmed in a degradation of the health score. If so, the root cause of the issue can be found by exploring the faults, and how these get rolled up in the larger model. Health scores also provide a real-time correlation in the event of a failure scenario, immediately providing feedback about which tenants, applications, and EPGs are impacted by that failure.

As a day-to-day operation, system administrators must monitor the health score as an ongoing activity, and resolve faults to improve the average health score of a given set of components over time.

The following figure displays an example how to analyze the degraded health score, and verify the root cause. If you navigate to the application profile, it has a health tab. In this tab is a tree that will show the various objects in a tree form to reveal the faults.
Verifying Health Score

Most objects in the model will have an associated health score, which can be found from the Dashboard or Policy tabs of the object from the GUI. Additionally, all health scores are instantiated from the healthInst class, and can be extracted using the Cisco APIC.

Additional References for Health Score

For additional information about health scores, see the *Cisco Application Centric Infrastructure Fundamentals Guide* at the following URL:

Using the Cisco NX-OS Style CLI

About Cisco NX-OS Style CLI

With the initial release of Cisco ACI, the majority of the configuration was done using either the Cisco APIC GUI or REST calls directly against the API. With the release of Cisco APIC release 1.2x, a NX-OS Style CLI was implemented to allow seasoned NX-OS users get a better feel for the capabilities of Cisco ACI, and how it maps to the existing NX-OS software. While the syntax is not an exact replication, the idea is to ease a user into a general understanding that the Cisco ACI fabric utilizes the same networking conventions that have been in place for years.

Prerequisites for Cisco NX-OS Style CLI

You must meet the following prerequisites to use the Cisco NX-OS Style CLI:

- The Cisco APIC cluster must be initialized and the Cisco APIC software should be at a minimum of release 1.2.

Guidelines and Limitations for Cisco NX-OS Style CLI

The following guidelines and limitations apply when using Cisco NX-OS Style CLI:

- From Cisco APIC Release 1.0 until Release 1.2, the default CLI was a Bash shell with commands to directly operate on managed objects (MOs) and properties of the Management Information Model (MIM). Beginning with Cisco APIC Release 1.2, the default CLI is a Cisco NX-OS style CLI. The object model CLI is available by typing the `bash` command at the initial CLI prompt.

- The NX-OS Style CLI is in the style of NX-OS, so there may be some syntactical differences.

- The NX-OS Style CLI and the Basic GUI utilize the same set of scripts to mask object complexity. As such, the NX-OS Style CLI has similar limitations when compared to the Basic GUI. It is recommended to pick one method of deployment and utilize that method indefinitely. Mixing and matching the deployment modes has potential to cause configuration overlap and overwrite unless great care is taken.

- When utilizing the NX-OS Style CLI, the output of `show run` represents the configuration within that specific submode. For example, `show run` within a tenant will only display all configurations within that tenant.

Verifying the Cisco NX-OS Style CLI

The Cisco NX-OS Style CLI can be accessed from the Cisco APIC. Once logged in, you are in the EXEC mode. The Cisco NX-OS Style CLI Global Configuration Mode can be accessed by entering `configure` to enter global configuration mode.
Configuration Examples for the Cisco NX-OS-Style CLI

For configuration for the Cisco NX-OS-style CLI, see the Cisco APIC Getting Started Guide at the following URL:


Additional References for the Cisco NX-OS-Style CLI

For more information about the NX-OS-style CLI, see the Cisco APIC NX-OS Style Command-Line Interface Configuration Guide at the following URL:


Upgrading the Fabric

Guidelines and Limitations for Adding a Switch

The following guidelines and limitations apply when adding a new switch:

• In the Cisco Cisco Application Centric Infrastructure (ACI) fabric, all fabric nodes should have the same software release version.

• The default firmware version can be found in the Application Policy Infrastructure Controller (APIC) GUI at Admin > Firmware > Fabric Node Firmware. Set the default firmware version to any.

• If you must return a switch for replacement using a Return Material Authorization (RMA), make sure to use the same node ID as previously configured so that the configured policies are pushed to the leaf switch.

• For more information about adding and removing fabric nodes, see the Operating Cisco Application Centric Infrastructure document at the following URL:


Guidelines and Limitations for Upgrading the Fabric

The following guidelines and limitations apply when upgrading the fabric:

• When you upload the software, use SCP or HTTP to download the software to the Cisco Application Policy Infrastructure Controller (APIC), and avoid directly uploading the firmware to the Cisco APIC if you do not have good network connectivity. The process will timeout if it takes too long.

• In production network, we recommend that you perform a manual upgrade rather than using a scheduler in the case that there are failures, and if further troubleshooting is needed.

• Do not upgrade or downgrade nodes that are part of a disabled configuration zone.
• Make sure that the Cisco APIC cluster is in fully fit status and that all devices are in the active status before upgrading.

• Make sure you have console access to all fabric nodes, in the case that case you must troubleshoot an issue.

• Make sure there are no outstanding faults before upgrading the fabric.

• Unless the release notes for the release specify otherwise, you can upgrade (or downgrade) the controllers before the switches, or upgrade (or downgrade) the switches before the controllers.

• Understand the supported downgrade path in the case that you are required to roll back the version.

• Make sure the controllers and switches are using the same software release.

• You can use a single firmware group for the upgrade process.

• When you create the maintenance group, verify the following items:
  • The vPC or active and standby pair of leaf switches are in two different groups so that while one of the switches is upgrading, the other switch can still pass the traffic.
  • Spine switches that are configured as MP-BGP router reflectors are in two different groups, otherwise you will lose external connectivity during the upgrade.

• Divide switches into two or more groups and upgrade one group at a time.

• A specific release, or a combination of releases, might have some limitations and recommendations for the upgrade or downgrade procedure. Look for any limitations and recommendations in the release notes for the specific release before upgrading or downgrading your Cisco Application Centric Infrastructure (ACI) fabric. If the release notes do not specify such limitations or recommendations, follow the guidelines to upgrade or downgrade your Cisco ACI fabric.

• Monitor the system faults to look for troubleshooting issues, and resolve any issues immediately.

• Verify that the Cisco APIC cluster is fully fit after the upgrade, before upgrading the spine switches and leaf switches.

• In the Run Mode field, choose the Pause only Upon Upgrade Failure radio button if it is not already chosen. This is the default mode.

• The default concurrent cap in a group is 20. This cap limits how many switches can go down simultaneously. You can increase the cap through a policy configuration.

• Verify that each maintenance group for the spine and leaf switches return to the active state after the upgrade.

Additional References for Upgrading the Fabric

For more details about upgrading the fabric, see the Cisco APIC Management, Installation, Upgrade, and Downgrade Guide at the following link:

Snapshot and Configuration Rollback

About Snapshot and Configuration Rollback

In earlier Cisco APIC releases, you could back up and restore your system configuration by exporting and importing configurations to and from external devices.

Beginning with Cisco APIC Release 1.2(1m), the snapshot and configuration rollback feature is available. It enables you to more easily revert to a previous configuration state, effectively rolling back any configuration changes that were made to a snapshot that was saved earlier.

Guidelines and Limitations when Using Snapshot and Rollback

The following guidelines and limitations apply when using snapshot and rollback features:

• By default, a snapshot for the entire fabric is taken, but you can also select snapshots of individual tenants if you desire.

• You can choose to store the snapshot either locally in Cisco APIC or on a remote server. If stored locally, the files will be synchronized across all Cisco APICs.

• You can import a file from a remote location, and save it as a snapshot.

• Snapshots can be created on a regular basis.

• Snapshots cannot be renamed.

• Only users with administrator privileges can perform snapshots.

• To rollback, two different snapshots are required to be operating. Cisco APIC will calculate the difference between the snapshots and apply the opposite of the difference.

• Only locally stored or imported snapshot files are supported for rollback.

• Export actions can also be scheduled to run at a future time or periodically. Import, export, and rollback jobs cannot run in parallel. If a job is already running, triggering a new job will fail.

Recommended Procedure for Snapshot and Rollback

When using the rollback feature, it is recommended that you first compare two snapshots within the GUI, and identify the configuration differences between them. Verify that the diff changes that will be applied are desired before proceeding with the rollback.

Configuration Example for Snapshot and Configuration Rollback

The following example displays how to create a snapshot in the Application Policy Infrastructure Controller (APIC) GUI.
Before you begin

You must have created a pre-determined location.

Procedure

<table>
<thead>
<tr>
<th>Step 1</th>
<th>On the menu bar, choose <strong>Admin &gt; Config Rollbacks</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the <strong>Work</strong> pane, you can perform actions to create a snapshot or rollback your configuration as desired.</td>
</tr>
</tbody>
</table>

## Verifying a Snapshot and Rollback Configuration

The following example command verifies the snapshot files before you rollback the files:

```bash
apic1# show snapshot files
File    : ce2_defaultOneTime-2016-05-10T19-00-18.tar.gz
Created : 2016-05-10T19:00:25.513-05:00
Root    : 
Size    : 180250

File    : ce2_defaultOneTime-2016-05-10T19-02-06.tar.gz
Created : 2016-05-10T19:02:13.018-05:00
Root    : 
Size    : 180118
```

## Additional References for Snapshot and Configuration Rollback

For additional information about snapshot and configuration rollback, see the *Cisco ACI Configuration Files: Import and Export* document at the following URL:


## Tags and Aliases

### About Tags and Aliases

Due to the interconnected nature of objects with respect to all relationships being formed in the Cisco APIC, an object cannot be renamed. This is because, if an object were to be renamed, a single name change would end up requiring a multitude of changes to the underlying model.

Therefore, there are constructs known as tags and aliases that allow a user to add some metadata against a group of objects for quick traversal using the API.

A tag allows objects to be grouped under a single string name, so that only the tag needs to be queried to find all objects associated with it.

An alias allows a new name over a specific object so that it can be referenced by that alias instead of the distinguished name or its actual name. As a result, each alias must be unique within a fabric.
Guidelines and Limitations for Tags and Aliases

The following guidelines and limitations apply for tags and aliases:

- Alias assignments must be unique to the fabric.
- Tags can be applied to multiple different objects.
- Aliases and tags have no functional impact on networking configuration.
- Labels are not the same as alias or tags. Labels are used for contract programming, and have a functional impact on network traffic.

Recommended Configuration Procedures for Tags and Aliases

A tag can be defined on multiple objects. In the Application Policy Infrastructure Controller (APIC) GUI, the Tags field and Alias field are located under the Policy tab in parts of the GUI where you can apply tags and define an alias.

Procedure

<table>
<thead>
<tr>
<th>Step 1</th>
<th>On the menu bar, choose Tenants &gt; All Tenants.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 2</td>
<td>In the Work pane, double-click the desired tenant's name.</td>
</tr>
<tr>
<td>Step 3</td>
<td>In the Navigation pane, choose Tenant tenant_name.</td>
</tr>
<tr>
<td>Step 4</td>
<td>In the Work pane, choose the Policy tab. The fields for Tags and Alias are included in the Work pane.</td>
</tr>
</tbody>
</table>

Verifying Tags and Aliases

The following information and examples are provided to verify tags and aliases:

- A configuration can be verified by performing a “Save as” with “all properties” and “subtree” on the tenant object, and viewing which attributes were set and under what class.

- An example of a tag definition is as follows:

  `<tagInst uid="0" status="" name="BP-Tag" monPolDn="uni/tn-common/monepg-default" modTs="2016-05-11T12:55:29.082-07:00" lcOwn="local" childAction="" rn="tag-BP-Tag"/>`

- An example of an alias definition is as follows:

  `<tagAliasInst status="" name="BP-TenantAlias" monPolDn="uni/tn-common/monepg-default" modTs="2016-05-11T12:55:47.421-07:00" lcOwn="local" childAction="" rn="alias"/>`

- An example of querying the tag using the API is as follows:
Get http://x.x.x.x/api/tag/BP-Tag.xml

• An example of a response is every object that has that tag associated is as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="2">
  <fvTenant childAction="" descr="" dn="uni/tn-ACI-BP" lcOwn="local" modTs="2016-05-10T09:06:37.165-07:00" monPolDn="uni/tn-common/monepg-default" name="ACI-BP" ownerKey="" ownerTag="" status="" uid="15374"/>
</imdata>
```

• An example of querying the alias using the API is as follows:

http://x.x.x.x/api/alias/BP-TenantAlias.xml

• An example of a response is the object to which the alias was assigned as follows:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="1">
  <fvTenant childAction="" descr="" dn="uni/tn-ACI-BP" lcOwn="local" modTs="2016-05-10T09:06:37.165-07:00" monPolDn="uni/tn-common/monepg-default" name="ACI-BP" ownerKey="" ownerTag="" status="" uid="15374"/>
</imdata>
```

### Additional References for Tags and Aliases

For more information about tags and aliases, see the *Cisco APIC REST API Configuration Guide* at the following URL:


### QuickStart in the Cisco APIC GUI

#### About QuickStart in the APIC GUI

In the APIC GUI, the QuickStart tab allows you to implement common tasks. It is essentially a wizard that is designed to guide you in a step-by-step fashion, through essential configuration tasks.

QuickStart is accessible using the APIC GUI. No additional configuration is required to access QuickStart.

In the APIC GUI, QuickStart is typically the first folder in the Navigation pane.

#### Prerequisites for QuickStart

You must meet the following prerequisites to use QuickStart:

- The Cisco Application Centric Infrastructure (ACI) fabric must be setup and accessible by HTTP or HTTPS.
Guidelines and Limitations for QuickStart

- QuickStart can be utilized for both Basic and Advanced configuration modes. In some cases it can be used for verifying the configuration.
- QuickStart is context sensitive. Depending on the location in the GUI, the QuickStart options may differ.
- QuickStart is not necessary for any of the configuration modes. For configuration, you can either use QuickStart or create and associate objects using the Advanced GUI.

Configuration Examples for QuickStart

The following procedure provides an example of using QuickStart in the Application Policy Infrastructure Controller (APIC) GUI for deploying an endpoint group.

Procedure

Step 1  On the menu bar, choose Fabric > Access Policies.
Step 2  In the Navigation pane, choose QuickStart.
Step 3  In the Work pane, click Configure an Interface, PC, and VPC.
        A list of existing switch interfaces displays. If you select a switch interface, the policy group name associated with that switch is displayed in the right pane.
Step 4  Click the dialog box link next to the Policy Group Name field to view the policies and Attached Entity Profile associated with that policy group.
Step 5  Click the dialog box link next to the Policy field to view the details of the associated object.
Step 6  Click the dialog box link next to the Attached Entity Profile field to view the details of the associated domains.