



Practice
Tests



Flash
Cards



Study
Planner



Video
Training

Official Cert Guide

Advance your IT career with hands-on learning

CCNP Data Center Application Centric Infrastructure

DCACI 300-620

About This eBook

ePUB is an open, industry-standard format for eBooks. However, support of ePUB and its many features varies across reading devices and applications. Use your device or app settings to customize the presentation to your liking. Settings that you can customize often include font, font size, single or double column, landscape or portrait mode, and figures that you can click or tap to enlarge. For additional information about the settings and features on your reading device or app, visit the device manufacturer's Web site.

Many titles include programming code or configuration examples. To optimize the presentation of these elements, view the eBook in single-column, landscape mode and adjust the font size to the smallest setting. In addition to presenting code and configurations in the reflowable text format, we have included images of the code that mimic the presentation found in the print book; therefore, where the reflowable format may compromise the presentation of the code listing, you will see a "Click here to view code image" link. Click the link to view the print-fidelity code image. To return to the previous page viewed, click the Back button on your device or app.

**CCNP Data Center
Application Centric
Infrastructure
DCACI 300-620
Official Cert Guide**

AMMAR AHMADI CCIE No. 50928

Cisco Press

CCNP Data Center Application Centric Infrastructure DCACI 300-620 Official Cert Guide Ammar Ahmadi

Copyright© 2021 Pearson Education, Inc.

Published by:

Cisco Press

Hoboken, NJ

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from the publisher, except for the inclusion of brief quotations in a review.

ScoutAutomatedPrintCode

Library of Congress Control Number: 2020948500

ISBN-13: 978-0-13-660266-8

ISBN-10: 0-13-660266-5

Warning and Disclaimer

This book is designed to provide information about the CCNP Implementing Cisco Application Centric Infrastructure DCACI 300-620 certification exam. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an “as is” basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at corpsales@pearsoned.com or (800) 382-3419.

For government sales inquiries, please contact governmentsales@pearsoned.com.

For questions about sales outside the U.S., please contact intlcs@pearson.com.

**Feedback Information At Cisco Press,
our goal is to create in-depth
technical books of the highest quality
and value. Each book is crafted with
care and precision, undergoing
rigorous development that involves
the unique expertise of members
from the professional technical
community.**

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Editor-in-Chief: Mark Taub **Alliances Manager, Cisco**
Press: Arezou Gol **Director, ITP Product Management:**
Brett Bartow **Executive Editor:** James Manly **Managing**
Editor: Sandra Schroeder **Development Editor:** Ellie Bru
Senior Project Editor: Tonya Simpson **Copy Editor:** Kitty
Wilson **Technical Editors:** Akhil Behl, Nikhil Behl **Editorial**
Assistant: Cindy Teeters **Cover Designer:** Chuti
Prasertsith **Composition:** codeMantra
Indexer: Erika Millen



Proofreader: Donna Mulder

Americas Headquarters

Cisco Systems, Inc.

San Jose, CA Asia Pacific Headquarters

Cisco Systems (USA) Pte. Ltd.

Singapore Europe Headquarters

Cisco Systems International BV Amsterdam,

The Netherlands Cisco has more than 200 offices worldwide.

Addresses, phone numbers, and fax 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: www.cisco.com/go/trademarks. 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. (1110R)

About the Author

Ammar Ahmadi, CCIE No. 50928, has nearly a decade of experience in data center design, implementation, optimization, and troubleshooting. He currently consults for Cisco Gold partner AHEAD INC, where he has been designing and supporting large-scale ACI fabrics since the early days of ACI. Occasionally, he breaks from design work to produce network modernization roadmaps or demonstrate the possibilities of software-defined networking (SDN) to customers.

Ammar also owns and operates Networks Reimagined LLC, which focuses on SDN enablement and training. He can be reached at ammar.ahmadi@networksreimagined.com.

About the Technical Reviewers

Akhil Behl, CCIE No. 19564, is a passionate IT executive with a key focus on cloud and security. He has more than 16 years of experience in the IT industry, working across several leadership, advisory, consultancy, and business development profiles with various organizations. His technology and business specialization includes cloud, security, infrastructure, data center, and business communication technologies. Currently he leads business development for cloud for a global systems integrator.

Akhil is a published author. Over the past few years, he has authored multiple titles on security and business communication technologies. He has contributed as technical editor to more than a dozen books on security, networking, and information technology. He has published several research papers in national and international journals, including *IEEE Xplore*, and presented at various IEEE conferences, as well as other prominent ICT, security, and telecom events. He is passionate about writing and mentoring.

He holds CCIE Emeritus (Collaboration and Security), Azure Solutions Architect Expert, Google Professional Cloud Architect, CCSK, CHFI, ITIL, VCP, TOGAF, CEH, ISM, CCDP, and many other industry certifications. He has a bachelor's degree in technology and an MBA.

Nikhil Behl, CCIE No. 23335, is a seasoned IT professional with exposure to a broad range of technologies. He has more than 15 years of experience working in the IT industry. He has worked in several ICT roles, including solutions architect, pre-sales lead, network architect, business

consultant, and CISCO TAC engineer, and he has worked with system integration and managed network services.

Nikhil has expertise in various technologies, including cloud, core networking, data center networking, software-defined networking, Wi-Fi, SD-WAN, and Software-Defined Access. He actively participates in several industry conferences and IT forums as a speaker.

Nikhil holds CCIE (Enterprise Infrastructure), Azure Solutions Architect Expert, Cisco SD-WAN Blackbelt, CCNP (Enterprise), CCDP, CCNA, CCDA, JNCIA (Junos), JNCIS, and many other industry leading certifications. He has a bachelor's degree in computer applications.

Dedication

I dedicate this book to my loving wife, Sophia, and my two children, Kiyana and Daniel. Sophia, your unrelenting support and patience made this book possible. I am forever grateful! Kiyana, you are full of life! You remind me every day that life is wonderful and that every moment should be cherished. Daniel, your nice big hugs have energized me at times when I really needed them. I look forward to spending more time with you and getting to know you more.

Acknowledgments

First, I would like to thank all the people who helped get this book to print. My special thanks go to Pearson product manager James Manly for his patience and understanding during this long process. I am also grateful to development editor Eleanor Bru, whose keen eye for detail has contributed tremendously to the quality of this book. Thank you, Brett Bartow, for the opportunity. And thanks to all the other people who contributed behind the scenes.

Second, I would like to thank all the people who have helped me grow in my career. I thank Peter Thompson, whom I saw as a mentor early in my career at Cisco. You helped me make several tough decisions that each greatly influenced my career. Also, thank you to Ryan Alt from AHEAD and John Rider from Cisco for allowing me the freedom to pick my projects. And thanks to all the unnamed folks at AHEAD who have made the past few years enjoyable. Thank you, Anthony Wilde, for showing me that you actually can have your cake and eat it, too. I would also like to acknowledge my wife, Sophia, for never letting me fall into the trap of complacency.

Finally, I would like to thank my mom, whose greatest desire has always been for me and my siblings to succeed, be happy, and achieve our dreams.

Contents at a Glance

Introduction

Part I Introduction to Deployment

Chapter 1 The Big Picture: Why ACI?

Chapter 2 Understanding ACI Hardware and Topologies

Chapter 3 Initializing an ACI Fabric

Chapter 4 Exploring ACI

Part II ACI Fundamentals

Chapter 5 Tenant Building Blocks

Chapter 6 Access Policies

Chapter 7 Implementing Access Policies

Chapter 8 Implementing Tenant Policies

Part III External Connectivity

Chapter 9 L3Outs

Chapter 10 Extending Layer 2 Outside ACI

Part IV Integrations

Chapter 11 Integrating ACI into vSphere Using VDS

Chapter 12 Implementing Service Graphs

Part V Management and Monitoring

Chapter 13 Implementing Management

Chapter 14 Monitoring ACI Using Syslog and SNMP

Chapter 15 Implementing AAA and RBAC

Part VI Operations

Chapter 16 ACI Anywhere

Part VII Final Preparation

Chapter 17 Final Preparation

Appendix A Answers to the “Do I Know This Already?”
Questions

Appendix B CCNP Data Center Application Centric
Infrastructure DCACI 300-620 Exam Updates

Glossary

Index

Online Elements

Appendix C Memory Tables

Appendix D Memory Tables Answer Key

Appendix E Study Planner

Glossary

Reader Services

Other Features

In addition to the features in each of the core chapters, this book has additional study resources on the companion website, including the following:

Practice exams: The companion website contains an exam engine that enables you to review practice exam questions. Use these to prepare with a sample exam and to pinpoint topics where you need more study.

Interactive exercises and quizzes: The companion website contains interactive hands-on exercises and interactive quizzes so that you can test your knowledge on the spot.

Glossary quizzes: The companion website contains interactive quizzes that enable you to test yourself on every glossary term in the book.

Video training: The companion website contains unique video samples from the author's complete video course.

To access this additional content, simply register your product. To start the registration process, go to www.ciscopress.com/register and log in or create an account.* Enter the product ISBN 9780136602668 and click Submit. After the process is complete, you will find any available bonus content under Registered Products.

*Be sure to check the box that you would like to hear from us to receive exclusive discounts on future editions of this product.

Contents

Introduction

Part I Introduction to Deployment

Chapter 1 The Big Picture: Why ACI?

“Do I Know This Already?” Quiz

Foundation Topics

Understanding the Shortcomings of Traditional Networks

Network Management

Scalability and Growth

Network Agility

Security

Network Visibility

Recognizing the Benefits of Cisco ACI

Network Management Touchpoints

Traffic Flow Optimizations

Scalability Optimizations

Programmability

Stateless Network

Multitenancy

Zero-Trust Security

Cross-Platform Integrations

New Architectural Possibilities

Integrated Health Monitoring and Enhanced Visibility

- Policy Reuse
- Exam Preparation Tasks
- Review All Key Topics
- Complete Tables and Lists from Memory
- Define Key Terms

Chapter 2 Understanding ACI Hardware and Topologies

- “Do I Know This Already?” Quiz
- Foundation Topics
- ACI Topologies and Components
 - Clos Topology
 - Standard ACI Topology
 - ACI Stretched Fabric Topology
 - ACI Multi-Pod Topology
 - ACI Multi-Site Topology
 - ACI Multi-Tier Architecture
 - Remote Leaf Topology
- APIC Clusters
 - APIC Cluster Scalability and Sizing
- Spine Hardware
 - First-Generation Spine Switches
 - Second-Generation Spine Switches
- Leaf Hardware
 - First-Generation Leaf Switches
 - Second-Generation Leaf Switches
- Exam Preparation Tasks
- Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Chapter 3 Initializing an ACI Fabric

“Do I Know This Already?” Quiz

Foundation Topics

Understanding ACI Fabric Initialization

Planning Fabric Initialization

Understanding Cabling Requirements

Connecting APICs to the Fabric

Initial Configuration of APICs

APIC OOB Configuration Requirements

Out-of-Band Versus In-Band Management

Configuration Information for Fabric
Initialization

Switch Discovery Process

Fabric Discovery Stages

Switch Discovery States

Initializing an ACI Fabric

Changing the APIC BIOS Password

Configuring the APIC Cisco IMC

Initializing the First APIC

Discovering and Activating Switches

Understanding Graceful Insertion and Removal
(GIR)

Initializing Subsequent APICs

Understanding Connectivity Following Switch
Initialization

Basic Post-Initialization Tasks

- Assigning Static Out-of-Band Addresses to Switches and APICs
- Applying a Default Contract to Out-of-Band Subnet
- Upgrading an ACI Fabric
- Understanding Schedulers
- Enabling Automatic Upgrades of New Switches
- Understanding Backups and Restores in ACI
- Making On-Demand Backups in ACI
- Making Scheduled Backups in ACI
- Taking Configuration Snapshots in ACI
- Importing Configuration Backups from Remote Servers
- Executing Configuration Rollbacks
- Pod Policy Basics
- Configuring Network Time Protocol (NTP) Synchronization
- Configuring DNS Servers for Lookups
- Verifying COOP Group Configurations

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Chapter 4 Exploring ACI

- “Do I Know This Already?” Quiz
- Foundation Topics
- ACI Access Methods
 - GUI

CLI

APIC CLI

Switch CLI

API

Management Access Modifications

Understanding the ACI Object Model

Learning ACI Through the Graphical User Interface

Exploring the Object Hierarchy by Using Visore

Why Understand Object Hierarchy Basics for DCACI?

Policy in Context

Integrated Health Monitoring and Enhanced Visibility

Understanding Faults

The Life of a Fault

Acknowledging Faults

Faults in the Object Model

Monitoring Policies in ACI

Customizing Fault Management Policies

Squelching Faults and Changing Fault Severity

Understanding Health Scores

Understanding Events

Squelching Events

Understanding Audit Logs

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Part II ACI Fundamentals

Chapter 5 Tenant Building Blocks

“Do I Know This Already?” Quiz

Foundation Topics

Understanding the Basic Objects in Tenants

Tenants

Predefined Tenants in ACI

VRF Instances

Bridge Domains (BDs)

Endpoint Groups (EPGs)

Application Profiles

The Pain of Designing Around Subnet Boundaries

BDs and EPGs in Practice

Configuring Bridge Domains, Application Profiles, and EPGs

Classifying Endpoints into EPGs

APIC CLI Configuration of Tenant Objects

Contract Security Enforcement Basics

Contracts, Subjects, and Filters

Contract Direction

Contract Scope

Zero-Trust Using EPGs and Contracts

Objects Enabling Connectivity Outside the Fabric

External EPGs

Layer 3 Outside (L3Out)

Tenant Hierarchy Review

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Chapter 6 Access Policies

“Do I Know This Already?” Quiz

Foundation Topics

Pools, Domains, and AAEPs

VLAN Pools

Domains

Common Designs for VLAN Pools and Domains

Challenges with Overlap Between VLAN Pools

Attachable Access Entity Profiles (AAEPs)

Policies and Policy Groups

Interface Policies and Interface Policy Groups

Planning Deployment of Interface Policies

Switch Policies and Switch Policy Groups

Profiles and Selectors

Configuring Switch Profiles and Interface Profiles

Stateless Networking in ACI

Bringing It All Together

Access Policies Hierarchy in Review

Access Policies and Tenancy in Review

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Chapter 7 Implementing Access Policies

“Do I Know This Already?” Quiz

Foundation Topics

Configuring ACI Switch Ports

 Configuring Individual Ports

 Configuring Port Channels

 Configuring Virtual Port Channel (vPC) Domains

 Configuring Virtual Port Channels

 Configuring Ports Using AAEP EPGs

 Implications of Initial Access Policy Design on Capabilities

Configuring Access Policies Using Quick Start Wizards

 The Configure Interface, PC, and VPC Wizard

 The Configure Interface Wizard

Additional Access Policy Configurations

 Configuring Fabric Extenders

 Configuring Dynamic Breakout Ports

 Configuring Global QoS Class Settings

 Configuring DHCP Relay

 Configuring MCP

 Configuring Storm Control

 Configuring CoPP

 Modifying BPDU Guard and BPDU Filter Settings

 Modifying the Error Disabled Recovery Policy

 Configuring Leaf Interface Overrides

 Configuring Port Channel Member Overrides

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Chapter 8 Implementing Tenant Policies

“Do I Know This Already?” Quiz

Foundation Topics

ACI Endpoint Learning

 Lookup Tables in ACI

 Local Endpoints and Remote Endpoints

 Understanding Local Endpoint Learning

 Unicast Routing and Its Impact on Endpoint Learning

 Understanding Remote Endpoint Learning

 Understanding the Use of VLAN IDs and VNIDs in ACI

 Endpoint Movements Within an ACI Fabric

 Understanding Hardware Proxy and Spine Proxy

 Endpoint Learning Considerations for Silent Hosts

 Where Data Plane IP Learning Breaks Down

 Endpoint Learning on L3Outs

 Limiting IP Learning to a Subnet

 Understanding Enforce Subnet Check

 Disabling Data Plane Endpoint Learning on a Bridge Domain

 Disabling IP Data Plane Learning at the VRF Level

- Packet Forwarding in ACI
 - Forwarding Scenario 1: Both Endpoints Attach to the Same Leaf
 - Understanding Pervasive Gateways
 - Forwarding Scenario 2: Known Destination Behind Another Leaf
 - Verifying the Traffic Path Between Known Endpoints
 - Understanding Learning and Forwarding for vPCs
 - Forwarding Scenario 3: Spine Proxy to Unknown Destination
 - Forwarding Scenario 4: Flooding to Unknown Destination
 - Understanding ARP Flooding
- Deploying a Multi-Tier Application
 - Configuring Application Profiles, BDs, and EPGs
 - Assigning Domains to EPGs
 - Policy Deployment Following BD and EPG Setup
 - Mapping EPGs to Ports Using Static Bindings*
 - Verifying EPG-to-Port Assignments*
 - Policy Deployment Following EPG-to-Port Assignment*
 - Mapping an EPG to All Ports on a Leaf
 - Enabling DHCP Relay for a Bridge Domain
- Whitelisting Intra-VRF Communications via Contracts
 - Planning Contract Enforcement
 - Configuring Filters for Bidirectional Application

Configuring Subjects for Bidirectional Application of Filters
Understanding Apply Both Directions and Reverse Filter Ports
Verifying Subject Allocation to a Contract
Assigning Contracts to EPGs
Understanding the TCP Established Session Rule
Creating Filters for Unidirectional Application
Configuring Subjects for Unidirectional Application of Filters
Additional Whitelisting Examples
Verifying Contract Enforcement
Understanding the Stateful Checkbox in Filter Entries
Contract Scopes in Review

Exam Preparation Tasks
Review All Key Topics
Complete Tables and Lists from Memory
Define Key Terms

Part III External Connectivity

Chapter 9 L3Outs

“Do I Know This Already?” Quiz
Foundation Topics
L3Out Fundamentals
 Stub Network and Transit Routing
 Types of L3Outs
 Key Functions of an L3Out

- The Anatomy of an L3Out
 - Planning Deployment of L3Out Node and Interface Profiles
 - Understanding L3Out Interface Types
 - Understanding L3Out Bridge Domains
 - Understanding SVI Encap Scope
 - Understanding SVI Auto State
 - Understanding Prerequisites for Deployment of L3Outs
 - L3 Domain Implementation Examples
 - Understanding the Need for BGP Route Reflection
 - Implementing BGP Route Reflectors
 - Understanding Infra MP-BGP Route Distribution
- Deploying L3Outs
 - Configuring an L3Out for EIGRP Peering
 - Deploying External EPGs
 - Verifying Forwarding Out an L3Out
 - Advertising Subnets Assigned to Bridge Domains via an L3Out
 - Enabling Communications over L3Outs Using Contracts
 - Deploying a Blacklist EPG with Logging
 - Advertising Host Routes Out an ACI Fabric
 - Implementing BFD on an EIGRP L3Out
 - Configuring Authentication for EIGRP
 - EIGRP Customizations Applied at the VRF Level
 - Configuring an L3Out for OSPF Peering
 - A Route Advertisement Problem for OSPF and EIGRP L3Outs

- Implementing BFD on an OSPF L3Out
- OSPF Customizations Applied at the VRF Level
- Adding Static Routes on an L3Out
- Implementing IP SLA Tracking for Static Routes
- Configuring an L3Out for BGP Peering
- Implementing BGP Customizations at the Node Level
- Implementing Per-Neighbor BGP Customizations
- Implementing BFD on a BGP L3Out
- Implementing BGP Customizations at the VRF Level
- Implementing OSPF for IP Reachability on a BGP L3Out
- Implementing Hot Standby Router Protocol (HSRP)
- IPv6 and OSPFv3 Support
- Implementing Route Control
 - Route Profile Basics
 - Modifying Route Attributes to All Peers Behind an L3Out
 - Modifying Route Attributes to a Specific Peer Behind an L3Out
 - Assigning Different Policies to Routes at the L3Out Level
 - Configuring Inbound Route Filtering in ACI
- Exam Preparation Tasks
 - Review All Key Topics
 - Complete Tables and Lists from Memory
 - Define Key Terms

Chapter 10 Extending Layer 2 Outside ACI

“Do I Know This Already?” Quiz

Foundation Topics

Understanding Network Migrations into ACI

Understanding Network-Centric Deployments

Understanding Full-Mesh Network-Centric Contracts

Understanding Any EPG

Understanding Preferred Group Members

Disabling Contract Enforcement at the VRF Instance Level

Flooding Requirements for L2 Extension to Outside Switches

Understanding GARP-Based Detection

Understanding Legacy Mode

Endpoint Learning Considerations for Layer 2 Extension

Preparing for Network-Centric Migrations

Implementing Layer 2 Connectivity to Non-ACI Switches

Understanding EPG Extensions

Understanding Bridge Domain Extensions

Comparing EPG Extensions and BD Extensions

Implementing EPG Extensions

Implementing L2Outs

Migrating Overlapping VLANs into ACI

Understanding ACI Interaction with Spanning Tree Protocol

Remediating Against Excessive Spanning Tree Protocol TCNs

Configuring MST Instance Mappings in ACI

Understanding Spanning Tree Protocol Link Types

Using MCP to Detect Layer 2 Loops

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Part IV Integrations

Chapter 11 Integrating ACI into vSphere Using VDS

“Do I Know This Already?” Quiz

Foundation Topics

Understanding Networking in VMware vSphere

Understanding vSphere Standard Switches

Understanding vSphere Distributed Switches

Understanding vSphere System Traffic

Impact of vCenter Failure on Production Traffic

Understanding Port Bindings in vSphere

Understanding Teaming and Failover Policies

Understanding VMM Integration

Planning vCenter VMM Integrations

What Happens After VDS Deployment?

Understanding Immediacy Settings

Connecting ESXi Servers to the Fabric

Configuring Connectivity to ESXi in UCS Domains

Integrating ACI into vSphere Using VDS
Prerequisites for VMM Integration with vSphere VDS
Configuring a VMM Domain Profile
Adding ESXi Hosts to a VDS
Pushing EPGs to vCenter as Distributed Port Groups
Assigning VMs to Distributed Port Groups
Less Common VMM Domain Association Settings
Enhanced LACP Policy Support
Exam Preparation Tasks
Review All Key Topics
Complete Tables and Lists from Memory
Define Key Terms

Chapter 12 Implementing Service Graphs

“Do I Know This Already?” Quiz
Foundation Topics
Service Graph Fundamentals
Service Graphs as Concatenation of Functions
Service Graph Management Models
Understanding Network Policy Mode
Understanding Service Policy Mode
Understanding Service Manager Mode
When to Use Service Graphs
Choosing an L4-L7 Services Integration Method
Understanding Deployment Modes and the Number of BDs Required

Deploying Service Graphs for Devices in GoTo Mode

Deploying Service Graphs for Devices in GoThrough Mode

Deploying Service Graphs for One-Arm Load Balancers

Understanding Route Peering

Understanding Dynamic Endpoint Attach

Understanding Bridge Domain Settings for Service Graphs

Understanding Service Graph Rendering

Service Graph Implementation Workflow

Importing Device Packages

Identifying L4–L7 Devices to the Fabric

Creating Custom Function Profiles

Configuring a Service Graph Template

Configuring Device Selection Policies

Applying a Service Graph Template

Configuring Additional Service Graph Parameters

Monitoring Service Graphs and Devices

Service Graph Implementation Examples

Deploying an Unmanaged Firewall Pair in a Service Graph

Deploying Service Graphs for a Firewall in Managed Mode

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Part V Management and Monitoring

Chapter 13 Implementing Management

“Do I Know This Already?” Quiz

Foundation Topics

Configuring Management in ACI

 Understanding Out-of-Band Management Connectivity

 Understanding In-Band Management Connectivity

 Deploying In-Band and OOB Management Side by Side

 Configuring In-Band Management

 Configuring Access Policies for APIC In-Band Interfaces

 Configuring the In-Band Management Bridge Domain

 Configuring In-Band Management IP Addressing

 Optionally Extending the In-Band Network Out of the Fabric

 Optionally Setting Up Additional Connectivity

 Whitelisting Desired Connectivity to and from an In-Band EPG

 Evaluating APIC Connectivity Preferences

 Out-of-Band Management Contracts in Review

Exam Preparation Tasks

Review All Key Topics

Memory Tables

Define Key Terms

Chapter 14 Monitoring ACI Using Syslog and SNMP

“Do I Know This Already?” Quiz

Foundation Topics

Understanding System Messages

Forwarding System Messages to Syslog Servers

 Apply Necessary Contracts to Allow Syslog Forwarding

 Configuring Syslog Monitoring Destination Groups

 Configuring Syslog Sources for Desired Monitoring Policies

 Verify Syslog Forwarding to Desired Syslog Servers

Using SNMP in ACI

 ACI Support for SNMP

 ACI SNMP Configuration Caveats

Configuring ACI for SNMP

 Apply Necessary Contracts for SNMP

 Associate an SNMP Policy with a Pod Policy

 Associate SNMP Contexts with Desired VRF Instances

 Configure SNMP Monitoring Destination Groups

 Configure SNMP Sources for All Desired Monitoring Policies

 Verify SNMP Forwarding to Desired SNMP Servers

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Chapter 15 Implementing AAA and RBAC

“Do I Know This Already?” Quiz

Foundation Topics

Implementing Role-Based Access Control (RBAC)

 Understanding Security Domains

 Understanding Privileges and Roles

 Creating Local Users and Assigning Access

 Tweaking Roles and User Access

 Custom RBAC Rules

 A Common RBAC Pitfall

Integrating with External AAA Servers

 Configuring ACI for TACACS+

 Configuring ISE to Authenticate and Authorize
 Users for ACI

 Expected Cisco AV Pair Formatting for ACI

 Configuring ACI for RADIUS

 Configuring ACI for LDAP

 AAA Authentication Policy Settings

 Regaining Access to the Fabric via Fallback
 Domain

Exam Preparation Tasks

Review All Key Topics

Complete Tables and Lists from Memory

Define Key Terms

Part VI Operations

Chapter 16 ACI Anywhere

“Do I Know This Already?” Quiz

Foundation Topics

ACI Multi-Site Fundamentals

Interconnecting ACI Fabrics with ACI Multi-Site

New ACI Multi-Site Constructs and Configuration Concepts

Locally Governed Versus MSO-Governed Configurations

Schemas and Templates in Practice

Building Primary and Disaster Recovery Data Centers with ACI

Centralized Orchestration and Management of Multiple Fabrics

Tweaking Broadcast and Stretch Settings on a Per-BD Basis

Cross-Data Center Ingress Routing Optimizations

Simultaneous or Independent Policy Deployment to Sites

Building Active/Active Data Centers with ACI

VMM Integrations Applicable to Multiple Data Centers

Stateful-Services Integration in ACI Multi-Pod and Multi-Site

Extending ACI to Remote Locations and Public Clouds

Extending ACI into Public Clouds with ACI Multi-Site

Extending ACI into Bare-Metal Clouds with vPod

Integrating Remote Sites into ACI Using Remote Leaf Switches

Exam Preparation Tasks

Review All Key Topics

Memory Tables

Define Key Terms

Part VII Final Preparation

Chapter 17 Final Preparation

Getting Ready

Tools for Final Preparation

Pearson Cert Practice Test Engine and Questions on the Website

Accessing the Pearson Test Prep Software Online

Accessing the Pearson Test Prep Software Offline

Customizing Your Exams

Updating Your Exams

Premium Edition

Suggested Plan for Final Review/Study

Summary

Appendix A Answers to the “Do I Know This Already?” Questions

Appendix B CCNP Data Center Application Centric Infrastructure DCACI 300-620 Exam Updates

Glossary

Index

Online Elements

[Appendix C Memory Tables](#)

[Appendix D Memory Tables Answer Key](#)

[Appendix E Study Planner](#)

[Glossary](#)

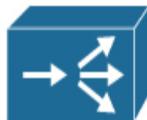
Icons Used in This Book



Cisco Nexus 7000



Cisco Nexus 5000



Local Director



Pix Firewall



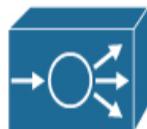
Router



File Server



Firewall



Application
Control Engine



Cisco Nexus 9000
in NX-OS Mode



APIC Controller



WWW Server



Terminal



Cloud



Detector



Switch

Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ([{ }]) indicate a required choice within an optional element.

Introduction

Welcome to the brave new world of Cisco ACI! This book strives to help you to:

- Understand the benefits of Cisco ACI and unlock its often-untapped potential
- Gain the expertise necessary to design, deploy, and support single-pod ACI fabrics
- Pass the Implementing Cisco Application Centric Infrastructure DCACI 300-620 exam.

The order of these three objectives is very important. An exam candidate who has an in-depth understanding of the fundamentals of a solution not only has an easier time on exam day but is also, arguably, a more capable engineer. That is why this book places an extraordinary amount of emphasis on the fundamentals of ACI rather than tips and tricks, corner-case scenarios, and platform-specific caveats.

This does not mean that this book is lacking in coverage of the DCACI blueprint. On the contrary, this book covers all the exam topics and then some. It does so with plain language and example after example of how particular features can be deployed and how they fit into the bigger picture of enabling ACI to be the data center SDN platform of the future.

Perspectives on the DCACI 300-620 Exam

In June 2019, Cisco announced that it was making substantial changes to certification products at all levels.

Cisco Application Centric Infrastructure (ACI) is a case in point for why these changes were necessary. Previous Cisco Certified Network Professional (CCNP) certifications followed a monolithic approach that necessitated major changes at both the CCNP and Cisco Certified Network Associate (CCNA) levels before a newer solution like ACI could be retrofitted into an overall curriculum. It commonly took several years for even immensely popular products (like ACI) to make it into the CCNP—and some never made it.

Newer Cisco certifications, on the other hand, take a more modular approach and encourage specialization in solutions most relevant to candidate job roles. If, for example, you are only interested in ACI, you can just take the DCACI 300-620 exam and obtain a specialist designation instead of a CCNA or CCNP. In the case of ACI, the Cisco certification evolution translates into greater depth of coverage without having content dispersed into a daunting number of exams alongside unrelated content.

One challenge that remains is that designing a certification covering all facets of a network product can require candidates to learn several thousand pages of content. This would unnecessarily discourage exam takers. Cisco has therefore divided coverage of ACI into two main exams:

- The DCACI 300-620 exam covers the fundamentals of ACI single-pod fabrics, such as endpoint learning, forwarding, management, monitoring, and basic integrations. In addition to being a specialization exam, the DCACI 300-620 exam also counts as a concentration toward the CCNP Data Center certification.
- The Implementing Cisco Application Centric Infrastructure—Advanced (300-630 DCACIA) exam addresses the implementation of more advanced ACI

architectures, such as ACI Multi-Pod and ACI Multi-Site. It also covers route leaking, advanced contract implementation, and service insertion via policy-based redirect (PBR).

The DCACI 300-620 exam addresses at least 70% of the concepts a typical ACI engineer deals with on a day-to-day basis and provides an excellent on ramp for engineers seeking to build the foundational knowledge necessary to implement the most complex of ACI designs.

As you might have noticed, one essential topic still missing from the blueprints of these two exams is network automation. Cisco has released a dedicated exam for data center automation that includes ACI, called the Automating and Programming Cisco Data Center Solutions (300-635 DCAUTO) exam. Therefore, this book does not cover network automation, opting instead to serve as a tool to help engineers build a solid foundation in ACI.

Who Should Read This Book?

This book has been written with you in mind!

For engineers new to ACI, this book attempts to demystify the complex language of ACI by using unambiguous wording and a wide range of examples. It includes detailed configuration steps and can even be used as a lab guide. This book recognizes ACI newcomers as a significant part of its target audience and has been written to be the most comprehensive and up-to-date book on ACI while also being the easiest to read.

For more advanced engineers who have experience with ACI but need a guide to prepare for the DCACI 300-620 exam or to address knowledge gaps, this book is comprehensive enough to address the topics on the exam while also taking

a look under the hood of ACI to enable these engineers to better appreciate how ACI works.

This book can also help network automation engineers build a solid foundation of ACI design and implementation concepts. Even though this book does not cover automation in ACI, it does address, in detail, how some of the most significant and often-used objects interact with one another.

This book is not an introduction to general networking and does expect readers to understand the basics of switching and routing. But this book does not assume that readers have any prior knowledge of ACI or even basic knowledge of data center overlay technologies. For this reason, this book can be used as a network engineer's first introduction to ACI.

The Companion Website for Online Content Review

All the electronic review elements, as well as other electronic components of the book, exist on this book's companion website.

To access the companion website, start by establishing a login at www.ciscopress.com and registering your book. To do so, simply go to www.ciscopress.com/register and enter the ISBN of the print book: 9780136602668. After you have registered your book, go to your account page and click the Registered Products tab. From there, click the Access Bonus Content link to get access to the book's companion website.

Note that if you buy the Premium Edition eBook and Practice Test version of this book from Cisco Press, your book will automatically be registered on your account page. Simply go to your account page, click the Registered Products tab,

and select Access Bonus Content to access the book's companion website.

How to Access the Pearson Test Prep (PTP) App

You have two options for installing and using the Pearson Test Prep application: a web app and a desktop app. To use the Pearson Test Prep application, start by finding the access code that comes with the book. You can find the code in these ways:

- **Print book:** Look in the cardboard sleeve in the back of the book for a piece of paper with your book's unique access code.
- **Premium edition:** If you purchase the Premium edition eBook and Practice Test directly from the Cisco Press website, the code will be populated on your account page after purchase. Just log in at www.ciscopress.com, click Account to see details of your account, and click the Digital Purchases tab.
- **Amazon Kindle:** For those who purchase a Kindle edition from Amazon, the access code will be supplied directly by Amazon.
- **Other bookseller eBooks:** Note that if you purchase an eBook version from any other source, the practice test is not included because other vendors to date have not chosen to vend the required unique access code.

Note

Do not lose the access code because it is the only means with which you can access the QA content with the book.

Once you have the access code, to find instructions about both the Pearson Test Prep web app and the desktop app,

1. follow these steps: Open this book's companion website.

Step 2. Click the **Practice Exams** button.

Step 3. Follow the instructions listed there for installing the desktop app and for using the web app.

If you want to use the web app only at this point, just navigate to www.pearsontestprep.com, establish a free login if you do not already have one, and register this book's practice tests using the access code you just found. The process should take only a couple of minutes.

Note

Amazon eBook (Kindle) customers: It is easy to miss Amazon's email that lists your Pearson Test Prep access code. Soon after you purchase the Kindle eBook, Amazon should send an email; however, the email uses very generic text and makes no specific mention of PTP or practice exams. To find your code, read every email from Amazon after you purchase the book. Also do the usual checks for ensuring your email arrives, like checking your spam folder.

Note

Other eBook customers: As of the time of publication, only the publisher and Amazon supply Pearson Test Prep access codes when you purchase their eBook editions of this book.

How This Book Is Organized

Although this book could be read cover-to-cover, it is designed to be flexible and allow you to easily move between chapters and sections of chapters to cover just the material that you are interested in learning. [Chapters 1](#) through [16](#) cover topics that are relevant to the DCACI 300-620 exam:

- **Chapter 1, “The Big Picture: Why ACI?”**: This chapter describes some of the challenges inherent in traditional network switches and routers and how ACI is able to solve these challenges.
- **Chapter 2, “Understanding ACI Hardware and Topologies”**: This chapter addresses the prominent ACI topologies in use today as well as ACI hardware platforms.
- **Chapter 3, “Initializing an ACI Fabric”**: This chapter covers planning parameters that are important for fabric initialization, the fabric initialization process itself, and some common post-initialization tasks, such as assignment of static out-of-band IP addresses to ACI nodes as well as making fabric backups and restoring configurations.
- **Chapter 4, “Exploring ACI”**: This chapter explores ACI access methods, the ACI object model, and some basic fabric health monitoring and fault management concepts.
- **Chapter 5, “Tenant Building Blocks”**: This chapter examines from a conceptual viewpoint the various objects present under the tenant hierarchy and how they relate to one another.
- **Chapter 6, “Access Policies”**: This chapter examines the concepts behind configuration of switch downlinks

to servers, external switches, and routers. It also addresses how switch port configurations tie in with the tenant hierarchy.

- **Chapter 7, “Implementing Access Policies”:** This chapter focuses on configuration of individual switch ports, port channels, vPCs, and fabric extenders (FEX) down to servers, external switches, and routers.
- **Chapter 8, “Implementing Tenant Policies”:** This chapter covers endpoint learning and forwarding in ACI as well as deployment of multitier applications and the enforcement of contracts to whitelist data center communications.
- **Chapter 9, “L3Outs”:** This chapter examines implementation of ACI route peering with outside Layer 3 devices as well as inbound and outbound route filtering.
- **Chapter 10, “Extending Layer 2 Outside ACI”:** This chapter addresses ACI Layer 2 connectivity with non-ACI switches and interaction with Spanning Tree Protocol. It also provides basic coverage of network migrations into and out of ACI.
- **Chapter 11, “Integrating ACI into vSphere Using VDS”:** This chapter addresses implementation of the most popular ACI integration and why it is important.
- **Chapter 12, “Implementing Service Graphs”:** This chapter tackles the introduction of firewalls and load balancers into ACI fabrics using service graphs.
- **Chapter 13, “Implementing Management”:** This chapter revisits the topic of in-band and out-of-band management in ACI and dives into the implementation of in-band management.

- **Chapter 14, “Monitoring ACI Using Syslog and SNMP”:** This chapter covers how ACI can forward faults and other monitoring information to syslog or SNMP servers.
- **Chapter 15, “Implementing AAA and RBAC”:** This chapter dives into role-based access control and how multitenancy can be enforced from a management perspective.
- **Chapter 16, “ACI Anywhere”:** This chapter provides a primer on additional ACI solutions within the ACI portfolio, including ACI Multi-Pod and ACI Multi-Site, which allow extension of ACI policies between data centers, between remote locations, and between public clouds.

How to Use This Book

The questions for each certification exam are a closely guarded secret. However, Cisco has published exam blueprints that list the topics you must know to *successfully* complete the exams. [Table I-1](#) lists the exam topics listed in the DCACI 300-620 exam blueprint along with a reference to the book chapter that covers each topic. These are the same topics you should be proficient in when designing and implementing ACI fabrics in the real world.

Table I-1 CCNP DCACI 300-620 Exam Topics and Chapter References

Exam Topic	Chapter(s) in Which Topic Is Covered

Exam Topic	Chapter(s) in Which Topic Is Covered
1.0 ACI Fabric Infrastructure	
1.1 Describe ACI topology and hardware	2
1.2 Describe ACI Object Model	4
1.3 Utilize faults, event record, and audit log	4
1.4 Describe ACI fabric discovery	3
1.5 Implement ACI policies 1.5.a access 1.5.b fabric	5, 6, 7

Exam Topic	Chapter(s) in Which Topic Is Covered
<p>1.6 Implement ACI logical constructs</p> <p>1.6.a tenant</p> <p>1.6.b application profile</p> <p>1.6.c VRF</p> <p>1.6.d bridge domain (unicast routing, Layer 2 unknown hardware proxy, ARP flooding)</p> <p>1.6.e endpoint groups (EPG)</p> <p>1.6.f contracts (filter, provider, consumer, reverse port filter, VRF enforced)</p>	5, 8, 9, 10
2.0 ACI Packet Forwarding	
2.1 Describe endpoint learning	8

Exam Topic	Chapter(s) in Which Topic Is Covered
2.2 Implement bridge domain configuration knob (unicast routing, Layer 2 unknown hardware proxy, ARP flooding)	8
3.0 External Network Connectivity	
3.1 Implement Layer 2 out (STP/MCP basics)	10
3.2 Implement Layer 3 out (excludes transit routing and VRF route leaking)	9
4.0 Integrations	
4.1 Implement VMware vCenter DVS integration	11
4.2 Describe resolution immediacy in VMM	11

Exam Topic	Chapter(s) in Which Topic Is Covered
4.3 Implement service graph (managed and unmanaged)	12
5.0 ACI Management	
5.1 Implement out-of-band and in-band	3 , 13
5.2 Utilize syslog and snmp services	14
5.3 Implement configuration backup (snapshot/config import export)	3
5.4 Implement AAA and RBAC	15
5.5 Configure an upgrade	3
6.0 ACI Anywhere	

Exam Topic	Chapter(s) in Which Topic Is Covered
6.1 Describe multipod	16
6.2 Describe multisite	16

Each version of the exam may emphasize different topics, and some topics are rather broad and generalized. The goal of this book is to provide comprehensive coverage to ensure that you are well prepared for the exam. Although some chapters might not address specific exam topics, they provide a foundation that is necessary for a clear understanding of important topics. Your short-term goal might be to pass this exam, but your long-term goal should be to become a qualified CCNP data center engineer.

It is important to understand that this book is a static reference, whereas the exam topics are dynamic. Cisco can and does change the topics covered on certification exams often.

This book should not be your only reference when preparing for the certification exam. You can find a wealth of information at Cisco.com that covers each topic in great detail. If you think you need more detailed information on a specific topic, read the Cisco documentation that focuses on that topic.

Note that as ACI features and solutions continue to evolve, Cisco reserves the right to change the exam topics without

notice. Although you can refer to the list of exam topics in Table I-1, you should check Cisco.com to verify the current list of topics to ensure that you are prepared to take the exam. You can view the current exam topics on any current Cisco certification exam by visiting the Cisco.com website and choosing Menu > Training & Events and selecting from the Certifications list. Note also that, if needed, Cisco Press might post additional preparatory content on the web page associated with this book at <http://www.ciscopress.com/title/9780136602668>. It's a good idea to check the website a couple weeks before taking the exam to be sure you have up-to-date content.

Figure Credit

[Figure 11-03](#): Screenshot of a VMkernel adapter with management services enabled © 2020 VMware, Inc [Figure 11-4](#): Screenshot of selecting Ephemeral - No Binding as the port binding type © 2020 VMware, Inc [Figure 11-5](#): Screenshot of teaming and failover settings for port groups © 2020 VMware, Inc [Figure 11-6](#): Screenshot of data center, cluster, and ESXi host hierarchy in vCenter © 2020 VMware, Inc [Figure 11-11](#): Screenshot of validating VDS creation in vCenter © 2020 VMware, Inc [Figure 11-12](#): Screenshot of navigating to the Add and Manage Hosts Wizard in vCenter © 2020 VMware, Inc [Figure 11-13](#): Screenshot of selecting add hosts © 2020 VMware, Inc

[Figure 11-14](#): Screenshot of clicking new hosts © 2020 VMware, Inc

[Figure 11-15](#): Screenshot of choosing the hosts to add on the Select New Hosts page © 2020 VMware, Inc [Figure 11-16](#): Screenshot of assigning uplinks to a VDS © 2020 VMware, Inc

[Figure 11-17](#): Screenshot of the Manage VMkernel Adapters page © 2020 VMware, Inc [Figure 11-18](#): Screenshot of the Manage VM Networking page © 2020 VMware, Inc

[Figure 11-19](#): Screenshot of confirming the addition of ESXi hosts to the VDS © 2020 VMware, Inc [Figure 11-22](#): Screenshot of verifying distributed port group generation in vCenter © 2020 VMware, Inc [Figure 11-23](#): Screenshot of reassigning a VM vNIC to a distributed port group © 2020 VMware, Inc [Figure 11-25](#): Screenshot of verifying the result of custom EPG naming and delimiter modification © 2020 VMware, Inc [Figure 11-26](#): Screenshot of verifying the result

of active uplinks and standby uplinks settings © 2020
VMware, Inc [Figure 11-28](#): Screenshot of assigning ESXi host
uplinks to a link aggregation group © 2020 VMware, Inc
[Figure 11-30](#): Screenshot of verifying distributed port group
mapping to uplinks © 2020 VMware, Inc

Part I: Introduction to Deployment

Chapter 1

The Big Picture: Why ACI?

This chapter covers the following topics:

Understanding the Shortcomings of Traditional Networks: This section discusses some of the challenges related to traditional data center networks.

Recognizing the Benefits of Cisco ACI: This section outlines how ACI addresses the major limitations of traditional networks.

The only way for businesses to be able to deploy and operationalize a technical solution in a way that takes full advantage of the benefits offered by the product is for engineers to develop a solid understanding of the product so that as they deploy it, they can keep its capabilities, corporate challenges, and industry challenges in mind.

This chapter provides a 10,000-foot view of Cisco Application Centric Infrastructure (ACI) and explores the reasons companies commonly deploy ACI. To this end, this chapter revisits the challenges that plagued more traditional networks and discusses how ACI addresses such challenges. The concepts outlined also set the stage for the more technical deep dives that follow in later chapters.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 1-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Questions.”](#)

Table 1-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding the Shortcomings of Traditional Networks	1-3
Recognizing the Benefits of Cisco ACI	4-10

Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

- 1.** Which of the following items contribute to network management complexity? (Choose all that apply.)

 - a.** Level of engineering expertise needed
 - b.** Open standards
 - c.** Number of managed endpoints
 - d.** Correlation of information across devices
- 2.** How many bits does the IEEE 802.1Q frame format use to define VLAN IDs, and what is the maximum number of VLAN IDs that can be used to segment Layer 2 traffic?

 - a.** 12 bits and 4094 VLAN IDs
 - b.** 10 bits and 1024 VLAN IDs
 - c.** 11 bits and 2048 VLAN IDs
 - d.** 24 bits and 8096 VLAN IDs
- 3.** True or false: Firewalls in traditional data centers focus primarily on securing east-west traffic.

 - a.** True
 - b.** False
- 4.** True or false: ACI uses VLANs internally to segment traffic.

 - a.** True
 - b.** False
- 5.** Which of the following solutions can be used as a single point of orchestration for multiple ACI fabrics?

 - a.** vPod
 - b.** ACI Multi-Pod
 - c.** The APIC GUI

- d. Multi-Site Orchestrator
- 6. True or false: In ACI, you can decommission a switch and then replace it in a few mouse clicks by reallocating a node ID.
 - a. True
 - b. False
- 7. Which of the following is the term for allowing all traffic except that which is denied by access lists and similar security mechanisms?
 - a. Blacklisting
 - b. Whitelisting
 - c. Multitenancy
 - d. Firewalling
- 8. True or false: Customers that avoid automation and orchestration cannot use ACI to achieve more agility.
 - a. True
 - b. False
- 9. Which ACI solution enables engineers to run additional control plane instances within a single ACI fabric?
 - a. Multisite
 - b. Configuration zones
 - c. Multipod
 - d. Security zones
- 10. Which one of the following items is not a reason to implement multitenancy?
 - a. Limiting the impact of configuration mishaps
 - b. Handing off network containers to business units

- c. Administrative separation of network resources
- d. Microsegmentation

Foundation Topics

Understanding the Shortcomings of Traditional Networks

In response to questions about the nature and purpose of Application Centric Infrastructure (ACI), engineers sometimes tend to answer simply that ACI is Cisco's response to the popularity of software-defined networking (SDN). Although it *is* controller based, agile, and highly programmable, ACI has actually been designed to address the broader range of industrywide headaches that have plagued traditional data center networks.

To better appreciate ACI, therefore, it's important to understand the context in which ACI was introduced to the market.

Network Management

Networks are complex to manage for a variety of reasons. The following are several of the common reasons for the complexity of network management:

- **Number of managed endpoints:** It is common for large enterprises to have hundreds of switches and routers in each data center. In addition to endpoints managed by network teams, storage nodes, compute nodes, firewalls, and load balancers further contribute to the number of endpoints that need to be individually managed. As the number of devices in a network

increases, the probability of configuration drift also increases, and so does the possibility of incorrect configurations going unnoticed for long periods of time. The number of devices also has a direct impact on the feasibility and effort needed for periodic end-to-end audits.

- **Correlating information between devices:** When complex issues occur and reactive mitigation is required, it can be cumbersome to correlate information across large numbers of switches, routers, compute nodes, virtualized environments, firewalls, and load balancers to reach resolutions.
- **Level of expertise:** Network maintenance is not always straightforward. A typical enterprise network leverages multiple networking vendors or at least multiple network operating systems. Each network is also configured differently. While it is common for network engineers to specialize in several platforms, it is not always common for all engineers to become subject matter experts in all areas of a corporate network. For this reason, interaction between multiple engineers is typically required, and a high level of expertise is therefore needed to keep networks performant.
- **Human error:** Even the most knowledgeable engineers may overlook some aspect of a network design or fat-finger a line of configuration. Any type of human error can be devastating and could lead to major downtime.
- **Differences in protocol implementations:** While open standards are often detailed, they are also very flexible, and a lot is left open to interpretation. Vendors therefore end up deploying solutions that have slight

differences across platforms, which can sometimes lead to network management headaches.

Scalability and Growth

Traditionally, networks have been built based on hierarchical designs that call for three layers. Within the data center, the three-tier design recommended by Cisco has consisted of an access layer, an aggregation layer, and a core layer. [Figure 1-1](#) shows a conceptual view of the three-tier data center architecture.

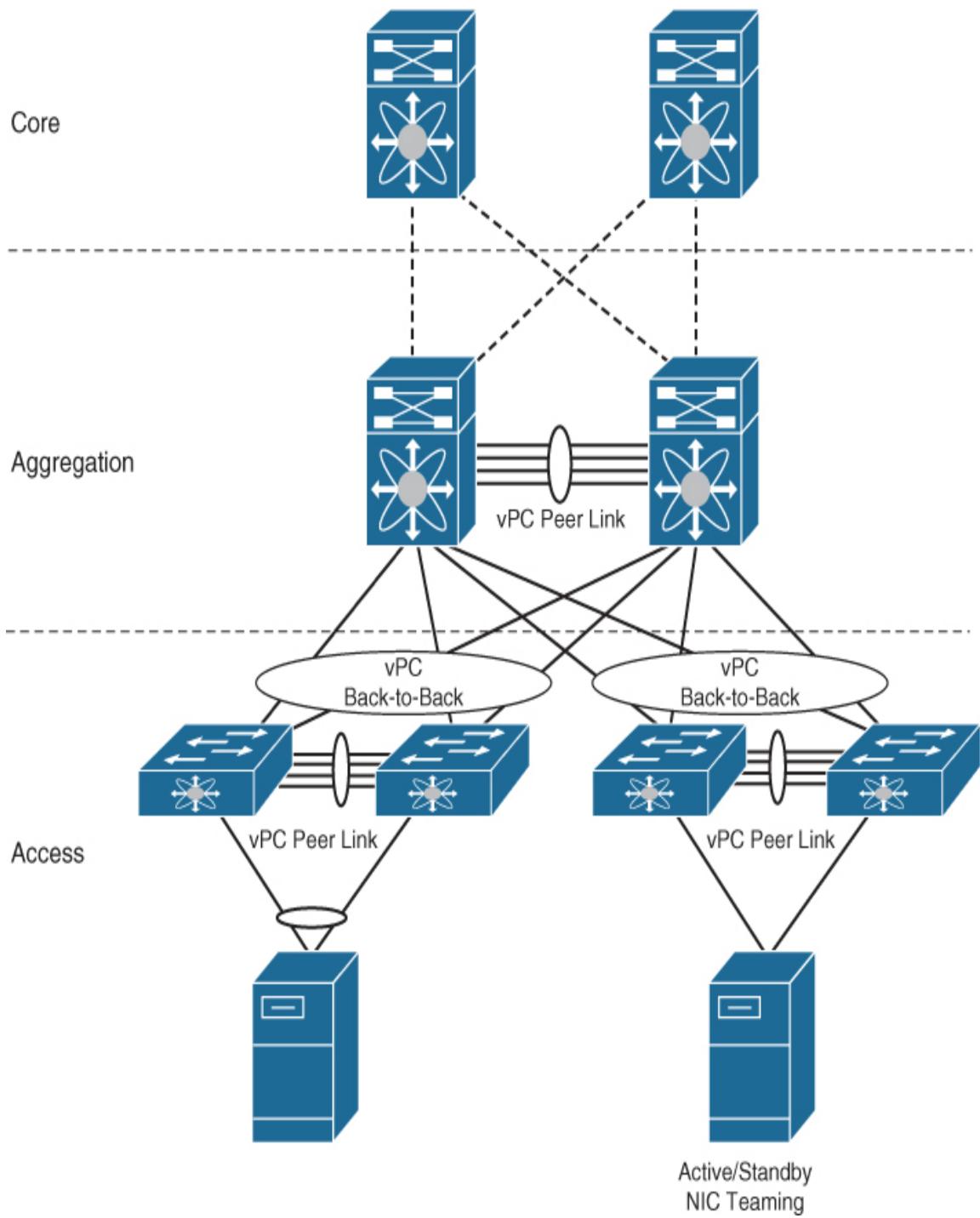


Figure 1-1 Conceptual View of the Traditional Three-Tier Data Center Architecture

Within the three-tier data center model, servers are typically housed in the access layer, where relatively low-cost

switches are used to perform line-rate intra-VLAN forwarding. A set of access switches connecting to a dedicated pair of aggregation layer switches forms a switch block.

The aggregation layer serves as the root of spanning tree for VLANs within data center switch blocks and also as a point for service aggregation. Inter-VLAN routing is performed at the aggregation layer, where default gateways are typically implemented using a first-hop redundancy protocol (FHRP), enabling an aggregation block to potentially sustain the loss of a single aggregation switch.

For aggregation layer downlinks to the access layer, Cisco recommends using Layer 2 virtual port channels (vPCs) or some other form of Multichassis EtherChannel (MEC) technology. Access layer switches are also often configured as vPC pairs to allow downstream servers to be dual-homed to the access layer using vPCs.

By definition, the data center core layer is intended to be formed of routed links and to support low-latency rapid forwarding of packets. The goal of the core layer is to interconnect the aggregation and access layers with other network zones within or outside the enterprise.

Historically, the three-tier data center hierarchical design enabled a substantial amount of predictability because use of aggregation switch blocks simplified the spanning-tree topology. The need for scalability and sometimes requirements for traffic separation pushed the three-tier model toward modularization, which in turn further increased predictability within the data center. [Figure 1-2](#) depicts a sample modular data center network design.

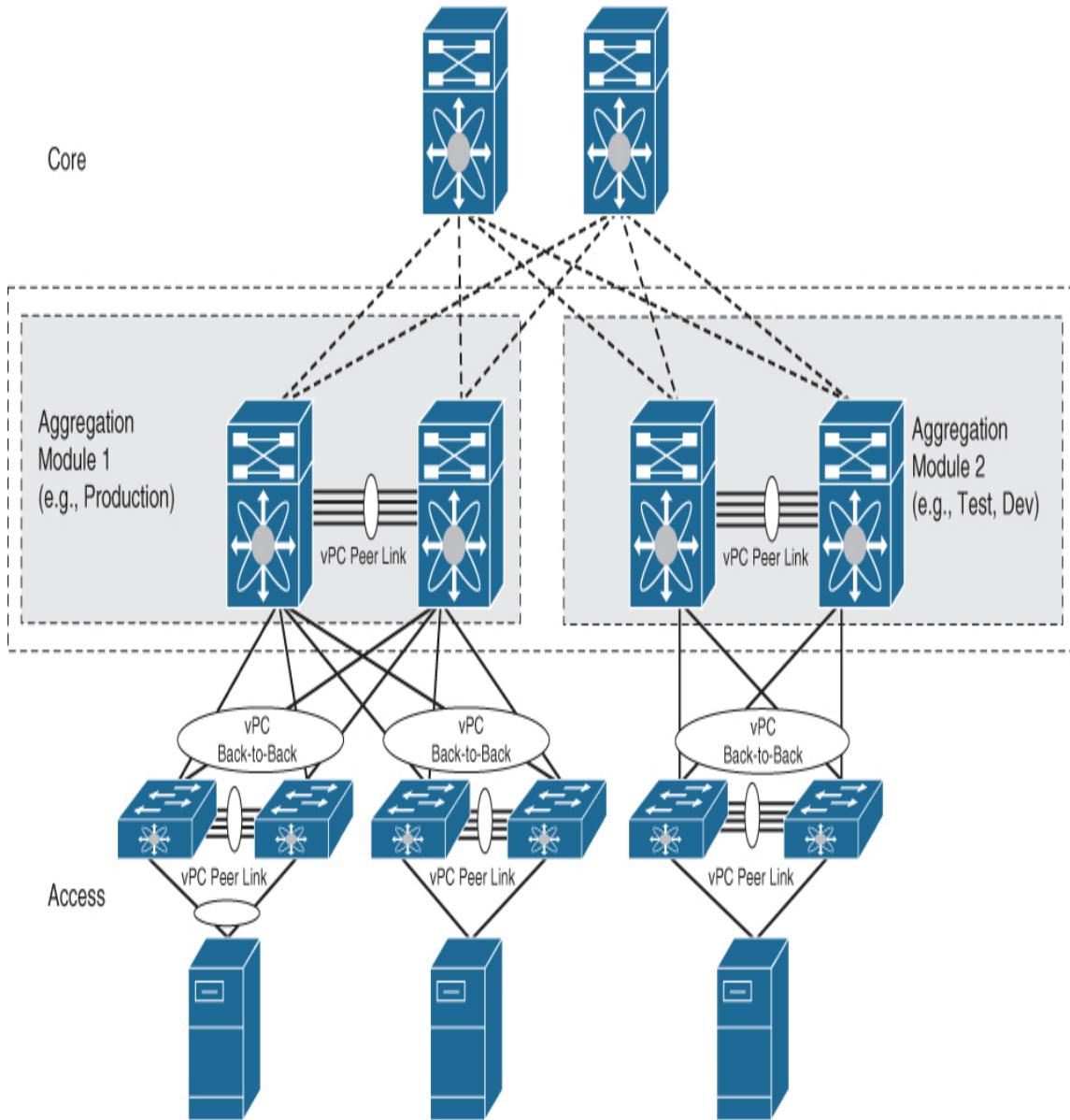


Figure 1-2 Example of Network Modularization Within the Data Center

The main challenge inherent in the three-tier model is that it is difficult to scale. To eliminate spanning-tree blocked links and maximize bandwidth between tiers, access and aggregation switches are typically paired into vPC domains, but each vPC domain can be formed using a maximum of two switches. Spanning-tree implications within the three-tier design prevented scaling out the aggregation layer

through addition of a third aggregation switch to a given switch block. In these traditional networks, to prevent bandwidth oversubscription within switch blocks from becoming a costly bottleneck, companies taking on data center redesign projects had to predict and often inflate their target-state network scalability requirements and purchase equipment accordingly. Knowing that applications tend to become more and more bandwidth hungry over time, companies that had deployed the 1 Gbps and 10 Gbps architectures of the time began to expect that vendors introduce newer and more flexible 40 Gbps and 100 Gbps architectures. The clear expectation in the industry was that application owners should be able to scale their applications knowing that the network could also be rapidly scaled to accommodate them.

An additional challenge that also guided the development of network solutions within data centers at the time was the growth in east-west traffic. The term *east-west traffic* refers to traffic between servers and is used to differentiate between traffic remaining in the data center and traffic to servers from the rest of the network (*north-south*). Server virtualization, distributed applications, and changes in the amount of data stored and replicated within data centers meant that traffic flows shifted in the east-west direction, and the network had to be able to support this growth. This shift coincided with demands for more agility and stability inside the data center, giving rise to a desire for subsecond failovers and a move away from spanning tree in favor of routed overlay networks.

This evolution continued with the emergence of technologies such as Cisco FabricPath, with which large enterprises were able to build extremely scalable topologies that did not suffer from spanning-tree blocked links and eliminated the need for modularization of data center switch

blocks. Using these new routed fabrics, companies also managed to optimize their data centers for east-west traffic flows. The popularity of routed fabrics and the industrywide embrace of a data center design approach that collapsed VLANs from all switch blocks within a data center into a single fabric and enabled any given VLAN to be available on any top-of-rack switch was enough for the industry to concede to routed fabrics being the way forward.

The fabric-based approach simplified the deployment of new switches and servers in data centers from a facilities standpoint and therefore decreased the time to deploy new services.

While these early fabrics were a step forward, each network switch within these fabrics still had to be configured independently. The early fabrics also did not address the issue of the number of managed network endpoints. Furthermore, the move of all VLANs into a single fabric also made apparent that some data center fabrics could be adversely impacted by IEEE 802.1Q limitations on the maximum number of VLANs.

Note

The IEEE 802.1Q standard that defines frame formats for supporting VLANs over Ethernet calls for the use of 12 bits to identify VLAN IDs; therefore, a maximum of 4096 VLANs are possible. With two VLANs (0 and 4095) reserved for system use, a total maximum of 4094 VLANs can be used to segment Layer 2 data plane traffic.

Although modularization is still desired in networks today, the general trend in large enterprise environments has been to move away from traditional architectures that revolve

around spanning tree toward more flexible and scalable solutions that are enabled by VXLAN and other similar Layer 3 overlay architectures.

Network Agility

The network underpins all IT services that modern businesses have come to rely on. For this reason, the network is almost always considered mission critical, and requests for changes to the network are understandably met with resistance.

The word *agility* in the context of IT refers to making configuration changes, deploying services, and generally supporting the business at the speed it desires; therefore, one company's definition of its expectations for agility will be different from that of another. In some environments, a network team may be considered agile if it can deploy new services in a matter of weeks. In others, agility may mean that business units in a company should be able to get applications to production or scale core services on demand through automation and orchestration with zero intervention from network engineers or even corporate IT.

Regardless of how a company decides to define agility, there is very little disagreement with the idea that network agility is vital to business success. The problem is that network agility has traditionally been hard to achieve.

Security

Switches and routers use access lists to enforce data plane security. However, companies have seldom been able to effectively leverage access lists on switches and routers for granular lockdown of server-to-server traffic.

Outside of server firewalls, the most common form of data plane security enforcement within data centers has been the use of physical firewalls, which are very effective in locking down north-south traffic flowing between security zones. Because firewalls are expensive and there is a latency impact associated with traffic inspection, redirecting east-west traffic that remains inside a single security zone is not always desirable.

[Figure 1-3](#) shows the typical firewall security zones deployed in most data centers today. Almost every data center has an Internet zone, a demilitarized zone (DMZ), and an inside security zone, but the exact number of security zones, the names associated with the zones, and the implementation details are very environment specific.

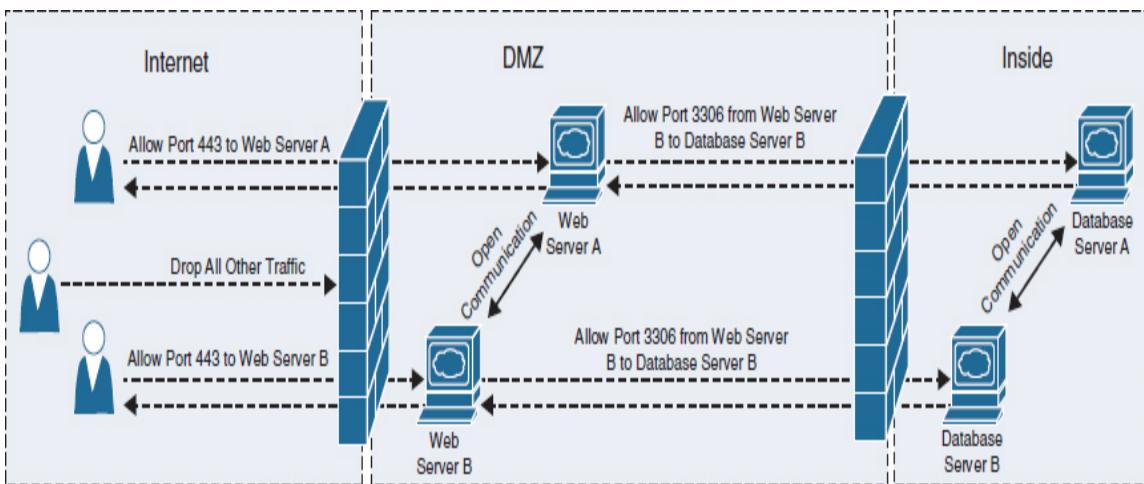


Figure 1-3 A Data Center Network Built Using Three Firewall Security Zones

The challenge with the traditional model of solely locking down north-south traffic via perimeter and inside firewalls is that once a server within a firewall security zone is compromised, other servers can then be impacted through lateral movement. For example, in [Figure 1-3](#), if Database Server A in the inside security zone were to become

infected, the malware could potentially move laterally to infect other servers, such as Database Server B, in the inside security zone without intervention from the north-south firewalls. Likewise, if a hacker were to compromise Web Server A, the hacker could then launch exploits laterally against Web Server B without firewall intervention. Even though Web Server A and Database Server A form an entirely different application instance than Web Server B and Database Server B, this type of lateral movement is common in traditional networks due to the complexities of enforcing and managing access lists on traditional switches and routers.

Network Visibility

Virtualization has muddied the boundaries between servers and the network. As more and more servers were deployed as virtual machines, troubleshooting of network connectivity issues became more complex because network teams only had visibility down to the hypervisor level and no visibility down to the VM level.

Even though solutions like Cisco's Nexus 1000v distributed virtual switch alleviate some visibility concerns, a lot of enterprises prefer to use hypervisor switches produced by the hypervisor vendors to simplify support and prevent interoperability issues.

Lack of end-to-end network visibility continues to be a concern for many network teams, especially now that container networking is becoming more popular.

Recognizing the Benefits of Cisco ACI

Fully embracing ACI requires a dramatic paradigm shift for most engineers. One motivation for outlining the ideal

target-state objectives for ACI-based networks is to give those transitioning into ACI a taste of the brave new world of data center networking. In addition, we want to provide very basic guideposts that may help inform engineers of blind spots as well as alternative approaches to ACI that may require different thinking in terms of design or configuration.

Some of the benefits listed in this section are inherent to ACI and some are products of good design and configuration practices. Not all of the benefits described in the following sections are exclusive to ACI. This section is not intended to provide a competitive comparison of data center network solutions available in the market today. It is also *not* intended to call out benefits in order of priority.

Network Management Touchpoints



Cisco ACI is deployed in a two-tier spine-and-leaf architecture in which every **leaf** connects to every **spine** in the topology. The leaf switches are the attachment points for all servers in the network. The spines serve to interconnect leaf switches at high speeds. The brain and central management point for the entire fabric is the **Cisco Application Policy Infrastructure Controller (APIC) cluster**, which is a set of (typically three) specialized servers that connect to leaf switches within the ACI fabric.

Figure 1-4 shows the components of an ACI fabric and their placement in a two-tier spine-and-leaf architecture.

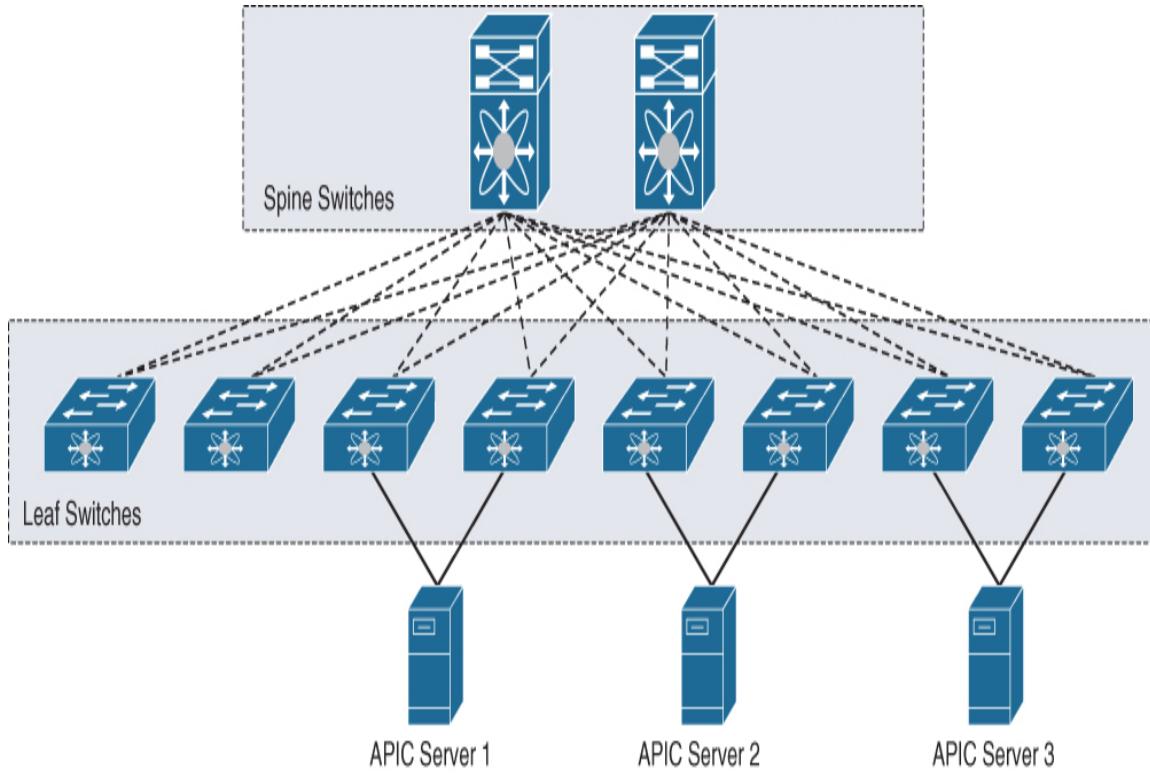


Figure 1-4 *Architecture and Components of an ACI Fabric*

In ACI, engineers do not directly configure switches. They send configurations to the APICs, which are the only configuration management points in the fabric, thereby reducing the number of touchpoints for configuration changes compared to traditional data center networks. Using APICs as a centralized component for fabric configuration also reduces the potential for human error because the APICs often identify when a new configuration conflicts with previously deployed configurations and may prevent the deployment of problematic configurations.

With ACI, engineers still have access to the individual switches for verification and troubleshooting purposes.

Traffic Flow Optimizations

Spine-and-leaf architectures optimize east-west traffic forwarding by ensuring that server-to-server traffic within a data center needs to traverse no more than a single spine and two leaf switches to reach the intended destination, making latency between servers more deterministic.

In addition to east-west traffic flow optimizations through the use of a spine-and-leaf physical topology, ACI has a number of data plane and control plane enhancements that lead to better traffic handling; these enhancements are covered in detail in [Chapter 8, “Implementing Tenant Policies.”](#)

Scalability Optimizations

Spine-and-leaf architectures that rely on routing protocols and not spanning tree enable a scale-out approach to growing the network. When more servers need to be deployed in a data center, you can expand ACI without outage and during regular work hours simply by adding new leaf switches. If oversubscription becomes a concern, you can introduce more leaf uplinks and/or spines into the fabric.

Another reason ACI fabrics are more scalable than traditional networks is that they are built around VXLAN, which uses 24 bits and provides over 16 million unique IDs. Note that ACI is VLAN aware, but VLANs in ACI are mostly used to encapsulate traffic egressing to external devices such as servers and routers as well as map inbound traffic entering the fabric to corresponding VXLAN network identifiers (VNIDs).

Programmability

ACI uses an advanced object-based model that allows for network constructs to be fully configured using an open representational state transfer (REST) API. In addition to providing this interface, ACI also provides a number of access methods that enable reading and manipulating of data.

In legacy network devices, APIs are an afterthought; in contrast, programmability is at the foundation of ACI. In fact, the ACI GUI programs the fabric using API calls.

The programmatic nature of ACI enables network agility; it allows for network and security engineers to script out changes and repeatable tasks to save time and enforce configuration standardization. For companies that seek to eliminate the need for manual changes through automation and orchestration, ACI programmability has even wider implications.

Stateless Network



An engineer who wants to configure an ACI fabric sends configurations to an APIC, which in turn configures the leaf and spine switches accordingly. Configurations are not associated with physical switches but with node IDs. A **node ID** is a logical representation of an ACI switch or APIC that can be associated with or disassociated from physical hardware. Because configurations are not bound to the physical devices, ACI hardware can be considered stateless.

In practice, statelessness enables engineers to look at switches as infrastructure that can be easily decommissioned and replaced with newer-model switches

faster and with minimal impact. Once a switch is replaced, an engineer assigns the node ID of the decommissioned switch to the new switch and the APIC and then configures the switch with all the same port assignments and configurations that were assigned to the old switch. Because platform-specific configuration parameters, such as interface names, are abstracted from node configurations as much as possible, and because it is the APICs and not the engineers that are tasked with interpreting how to deploy node configurations to physical switches, subsequent data center migrations can be dramatically expedited. Stateless networking, therefore, attempts to address the need for network agility.

By lowering data center migration times as well as the time to migrate off of faulty switches, stateless networking introduces additional cost savings that cannot be easily calculated. By saving costs in the long term and enabling an architecture that can be leveraged for decades to come, ACI frees up engineering time that can be refocused on more business-critical tasks, such as enforcement of enhanced security.

Multitenancy



Multitenancy is the ability to logically separate management as well as data plane forwarding of different logical environments that reside on top of common physical infrastructure.

Because multitenancy means different things to different people, it might be best to first examine characteristics of multitenant environments in general and then clarify how

multitenancy should be understood within the context of ACI.

A natural analogy for multitenancy in IT can be found in the operation of apartment buildings, and an analysis of how apartments enable multitenancy is therefore a great place to start.

An apartment building usually consists of several apartments, each of which is rented out to a different tenant. The apartment building is cost-effective for the owner because it uses shared infrastructure such as land, walls, parking lots, water pipes, natural gas infrastructure, and the electrical system.

A tenant who rents an apartment is given a key and has relative freedom to do as he or she pleases within the confines of the apartment. Renters can invite guests and can come and go as they please, regardless of the time of day or night.

Even though the electrical system is a shared infrastructure, a tenant does not need to worry too much about blowing the fuse on occasion due to use of high-amperage appliances. As a core design consideration, the building owner is expected to provide each tenant with core services and a reasonable amount of fault isolation. In other words, it is reasonable to expect that the loss of electricity within a single apartment should not impact the entire building or complex.

Tenants also have certain rights and responsibilities that govern their relationships with neighbors and the apartment owner as well as his or her representatives. A tenant has the right to privacy and understands that the apartment owner cannot enter his or her dwelling unless either there is an emergency or the tenant has been notified well in advance. Tenants also have a right to good living conditions. On the

other hand, the apartment owner expects that tenants pay rent in a timely fashion and avoid disruptive behavior or cause damage to the building.

An apartment owner typically signs a binding contract with a tenant. The law complements these contracts by enabling lawsuits against negligent owners and making evictions possible, where tenants may abandon their responsibilities.

All in all, the tenant/owner relationship benefits both parties. The owner performs maintenance where reasonable within apartments and is responsible for keeping the property grounds clean. The owner is also responsible for maintaining all shared infrastructure within the property. Since the tenants are not responsible for maintenance, they have more time to focus on the things that matter most to them.

Moving on to ACI, one can easily see that shared infrastructure is primarily a reference to the ACI fabric itself. But the idea of shared infrastructure also applies to servers and appliances residing within the fabric.

ACI inherits support for multitenancy through its use of the Multiprotocol Border Gateway Protocol Ethernet Virtual Private Network (MP-BGP EVPN) control plane for VXLAN. The MP-BGP EVPN control plane and other control plane instruments play a role similar to that of the law and the binding contract between the apartment owner and tenant in the previous example.

In addition to control plane dependencies, data plane and management plane aspects are at work to enable multitenancy capabilities similar to those outlined for apartments:

- **Data plane multitenancy:** As in the world of apartment rentals, in a network, there is a need to ensure that tenants remain exclusive and in controlled

environments and that issues within one tenant space do not impact other tenants. In ACI, a tenant is simply an object under which an administrator may configure one or more virtual routing and forwarding (VRF) instances. VRF instances segment traffic at Layer 3 by virtualizing the routing tables of routers. Because any given network subnet is always associated with a VRF instance and traffic originating within a subnet cannot traverse VRF boundaries unless intentionally leaked or advertised into the destination VRF, full data plane traffic segmentation is possible. ACI also enables multitenancy at Layer 2.



- **Management multitenancy:** Just as an apartment renter expects to have almost exclusive access to apartment keys and have the freedom to do as he or she pleases within the rented apartment, ACI multitenancy employs mechanisms to enable and enforce relevant users the freedoms they need within their tenant space. ACI introduces the concept of security domains to define the part of the object-based hierarchy within ACI that a user can access. ACI also controls the amount of access the user has by using role-based access control (RBAC).

Some common drivers for the implementation of multiple tenants in ACI are as follows:

- **Administrative separation:** When network administration for different applications or devices is handled by different teams within a company, a simple solution is to use multitenancy to carry over the same

administrative separation into ACI. A good example of this is hospitals that have a for-profit division as well as a nonprofit division. Each division may employ different engineers who manage different applications and servers, but they may be able to cut costs through the use of shared infrastructure.

- **Alignment with software development lifecycles:** There are a lot of use cases for leveraging tenants in software development. For example, in some environments, applications may be deployed in a tenant that mirrors production and serves as a staging ground for testing. Once the application is fully tested, it can then be moved into a production tenant without IP address changes.
- **Overall IT strategy:** IT is generally seen as a cost center. Some companies cross-charge departments for services as part of an effort to transform from a cost center to a provider of services to the business. In such cases, IT may intend to provide one or more tenants to each business unit to allow them to configure their own networking constructs, applications, and security so they can deploy IT services at the pace they expect. Decisions around tenant design often come down to overall IT strategy, and a wide range of reasons exist for tenant deployment that may be specific to an individual environment.
- **Partnerships, mergers, and acquisitions:** Let's say two companies enter into a partnership and need a common space to set up applications that will be owned jointly by both companies. In these types of scenarios, a tenant can be deployed with a security domain that employees from both companies are assigned to. Mergers and acquisitions create similar situations,

where granular control over networking environments may allow additional flexibility and agility.

- **Limiting fault domain sizes:** Similar to the case in which an apartment tenant blows a fuse and the expectation is that there will be no cascading effect leading to power outages across the entire apartment building, some customers use ACI multitenancy to limit and isolate fault domains from one another. Imagine that an IT organization creates a tenant for a series of applications that are business critical and makes a firm decision that management of that individual tenant will always remain under the control of the central IT organization. It then creates a separate tenant and hands off management of the tenant to a specific business unit that has specific requirements for more agile changes to the network in support of an important software development project. Let's say the IT team is more network savvy than the business unit in question. It understands that route overlaps between the new tenant and the business-critical tenant will have minimal considerations, but it is worried about the possibility of the new tenant owner incorrectly configuring subnets that propagate throughout the enterprise network and causing outages to other systems. Because the egress point for all VRF instances within the tenant is the default VRF instance on the data center core, IT has decided to implement very basic route filtering in the inbound direction on the core layer, allowing only subnets that are within a single supernet to be advertised out of ACI from the tenant in question. This basic solution, when combined with well-defined security domains and RBAC, can prevent any configuration mishap in the new development tenant from causing wider issues within the corporate network.

It is worth noting that multitenancy did exist in traditional networks using VRF instances and RBAC. However, multitenancy in ACI provides more granular control and is easier to configure.

Zero-Trust Security



In traditional data centers, north-south firewalls are the primary enforcers of security within the data centers. By default, switches and routers do not block any traffic and do little in terms of data plane security enforcement. If switches and routers use tools such as access lists to lock down data plane traffic, they are basically **blacklisting** certain traffic.



ACI is different from traditional switches and routers in that all traffic flows crossing VXLAN boundaries within the fabric are denied by default. Contracts define which subset of endpoints within ACI can communicate with one another. This default behavior in which traffic is dropped unless it is explicitly allowed via contracts is called whitelisting.

Whitelisting is more feasible in ACI than in traditional networks because the configuration of contracts is fundamentally different from that of CLI-based access lists, even though they are enforced nearly the same way.

The benefits of whitelisting can be better understood through analysis of a multitier application. [Figure 1-3](#), shown

earlier in the chapter, actually depicts two multitier applications, each consisting of a web tier and a database tier. When ACI and whitelisting come into the picture, the same traffic flows exist. However, only the minimally required ports between the application tiers are opened due to whitelisting rules, as depicted in [Figure 1-5](#). With this change, Web Server A can no longer talk to Web Server B. Likewise, Database Server A is also unable to communicate with Database Server B.

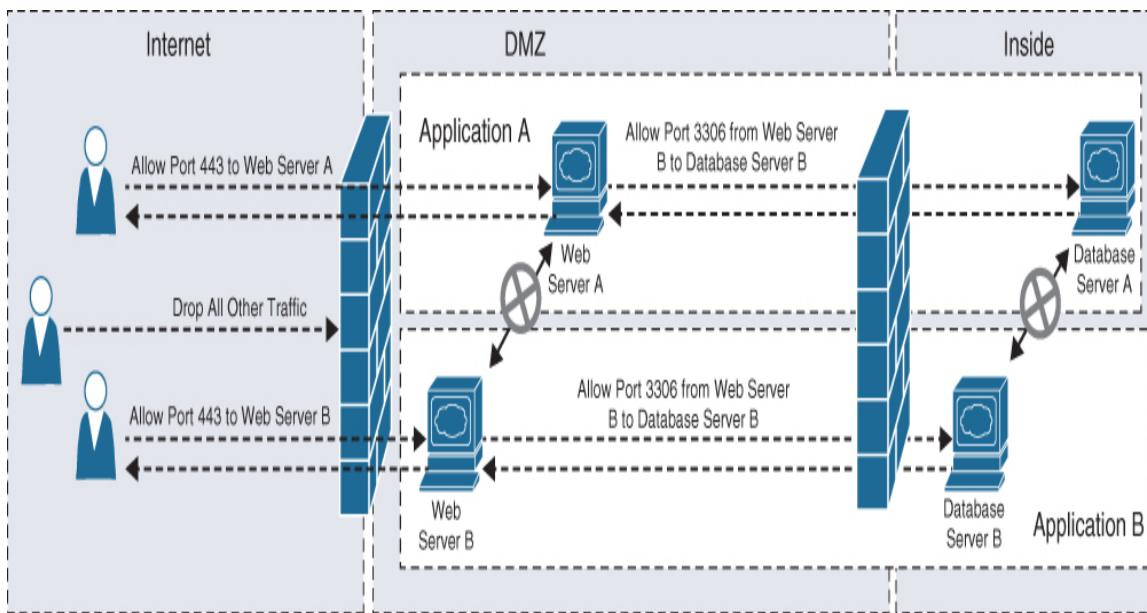


Figure 1-5 *Whitelisted Two-Tier Application*



ACI supports a **zero-trust security** architecture through whitelisting, allowing administrators to block non-essential communications other than those explicitly specified through contracts. Where more granular application-level inspection is also desired, ACI can redirect desired traffic to a firewall for inspection prior to forwarding to the destination server.

Cross-Platform Integrations

One of the challenges in traditional networks is cross-platform and cross-vendor visibility and integration. ACI has a large partner ecosystem and has integrations with an extensive number of vendors.

Although there is a wide variety of use cases for integrations with ACI, some of the more common include integrations with hypervisor environments and L4-L7 service insertion. ACI integration with VMware vCenter, for example, enables ACI to automatically push port groups into vSphere. One use case for L4-L7 service insertion is to selectively redirect traffic to a firewall or a pool of firewalls for more granular inspection of traffic.

Compared to legacy architectures, ACI cross-platform integrations provide a lot of benefits and primarily address the need for visibility and agility within data centers.

New Architectural Possibilities

In addition to attempting to address the challenges inherent in traditional data centers, ACI also creates new architectural possibilities.

ACI Multi-Pod, for instance, offers a way of segmenting a fabric into pods that each run separate control plane instances and is very effective in creating active/active data centers or staging equipment across data centers within a single campus environment.

ACI Multi-Site, on the other hand, is a solution that is enabled by Multi-Site Orchestrator (MSO), which can serve as a single point of policy orchestration across multiple ACI fabrics to enable flexible and even dual active/active architectures that allow for seamless policy movement

between data centers, among other things. It can also integrate with the Cisco Cloud APIC to allow homogenous security policy across on-premises data centers and public clouds, ensuring operational consistency and visibility across clouds.

Other ACI solutions include remote leaf and vPod.

Integrated Health Monitoring and Enhanced Visibility

ACI is a controller-based system, and by virtue of being a system and not a disparate collection of switches, ACI allows visibility into traffic flow and issues that may be causing packet loss and performance degradation.

ACI uses faults and health scores to determine the state of an overall system. Because of the deep visibility of the APIC controllers into the fabric, ACI is able to provide more analytics than regular monitoring tools such as syslog and SNMP typically provide. Integrated health monitoring and enhanced visibility within ACI typically translate to faster problem resolution and more proactive problem resolution.

Policy Reuse

While automation through scripting greatly helps agility, not all companies will embrace scripting and orchestration in the near future. Some companies simply expect to save as much time as possible without having to learn scripting. In such cases, policy reuse can help. For instance, by creating profiles for server interface configurations that can be instantiated anywhere in the data center, IT teams can reduce the amount of time needed to deploy new servers while also decreasing the possibility of configuration drift.

Exam Preparation Tasks

As mentioned in the section “How to Use This Book” in the Introduction, you have a couple of choices for exam preparation: [Chapter 17](#), “Final Preparation,” and the exam simulation questions in the Pearson Test Prep Software Online.

Review All Key Topics

Review the most important topics in this chapter, noted with the Key Topic icon in the outer margin of the page. [Table 1-2](#) lists these key topics and the page number on which each is found.



Table 1-2 Key Topics for Chapter 1

Key Topic Element	Description	Page Number
Paragraph	Lists the components of an ACI fabric	9
Paragraph	Explains how a node ID enables stateless networking	11

Paragraph	Defines multitenancy	11
List	Explains how data plane and management multitenancy work together to make multitenancy possible	12
Paragraph	Describes blacklisting	14
Paragraph	Describes whitelisting	14
Paragraph	Explains the high-level mechanisms ACI uses to establish zero-trust security	15

Complete Tables and Lists from Memory

There are no memory tables or lists in this chapter.

Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

blacklisting

whitelisting
node ID
APIC cluster
leaf
spine
zero-trust security
multitenancy

Chapter 2

Understanding ACI Hardware and Topologies

This chapter covers the following topics:

ACI Topologies and Components: This section describes the key hardware components and acceptable topologies for ACI fabrics.

APIC Clusters: This section covers available APIC hardware models and provides an understanding of APIC cluster sizes and failover implications.

Spine Hardware: This section addresses available spine hardware options.

Leaf Hardware: This section outlines the leaf platforms available for deployment in ACI fabrics.

This chapter covers the following exam topics:

- 1.1 Describe ACI topology and hardware
- 6.1 Describe Multi-Pod
- 6.2 Describe Multi-Site

ACI is designed to allow small and large enterprises and service providers to build massively scalable data centers using a relatively small number of very flexible topologies.

This chapter details the topologies with which an ACI fabric can be built or extended. Understanding supported ACI topologies helps guide decisions on target-state network architecture and hardware selection.

Each hardware component in an ACI fabric performs a specific set of functions. For example, leaf switches enforce security rules, and spine switches track all endpoints within a fabric in a local database.

But not all ACI switches are created equally. Nor are APICs created equally. This chapter therefore aims to provide a high-level understanding of some of the things to consider when selecting hardware.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 2-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Questions.”](#)

Table 2-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
ACI Topologies and Components	1–5

Foundation Topics Section	Questions
APIC Clusters	6
Spine Hardware	7, 8
Leaf Hardware	9, 10

Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

1. An ACI fabric is being extended to a secondary location to replace two top-of-rack switches and integrate a handful of servers into a corporate ACI environment. Which solution should ideally be deployed at the remote location if the deployment of new spines is considered cost-prohibitive and direct fiber links from the main data center cannot be dedicated to this function?
 - a. ACI Multi-Site
 - b. ACI Remote Leaf

- c. ACI Multi-Tier
 - d. ACI Multi-Pod
- 2. Which of the following is a requirement for a Multi-Pod IPN that is not needed in an ACI Multi-Site ISN?
 - a. Increased MTU support
 - b. OSPF support on last-hop routers connecting to ACI spines
 - c. End-to-end IP connectivity
 - d. Multicast PIM-Bidir
- 3. Which of the following connections would ACI definitely block?
 - a. APIC-to-leaf cabling
 - b. Leaf-to-leaf cabling
 - c. Spine-to-leaf cabling
 - d. Spine-to-spine cabling
- 4. Which of the following are valid reasons for ACI Multi-Site requiring more specialized spine hardware? (Choose all that apply.)
 - a. Ingress replication of BUM traffic
 - b. IP fragmentation
 - c. Namespace normalization
 - d. Support for PIM-Bidir for multicast forwarding
- 5. Which of the following options best describes border leaf switches?
 - a. Border leaf switches provide Layer 2 and 3 connectivity to outside networks.

- b.** Border leaf switches connect to Layer 4-7 service appliances, such as firewalls and load balancers.
- c.** Border leaf switches are ACI leaf switches that connect to servers.
- d.** Border leaf switches serve as the border between server network traffic and FCoE storage traffic.

6. Which of the following statements is accurate?

- a.** A three-node M3 cluster of APICs can scale up to 200 leaf switches.
- b.** Sharding is a result of the evolution of what is called horizontal partitioning of databases.
- c.** The number of shards distributed among APICs for a given attribute is directly correlated to the number of APICs deployed.
- d.** A standby APIC actively synchronizes with active APICs and has a copy of all attributes within the APIC database at all times.

7. Out of the following switches, which are spine platforms that support ACI Multi-Site? (Choose all that apply.)

- a.** Nexus 93180YC-EX
- b.** Nexus 9364C
- c.** Nexus 9736C-FX line card
- d.** Nexus 9396PX

8. Which of the following is a valid reason for upgrading a pair of Nexus 9336PQ ACI switches to second-generation Nexus 9332C spine hardware? (Choose all that apply.)

- a.** Namespace normalization for ACI Multi-Site support
- b.** Support for 40 Gbps leaf-to-spine connectivity
- c.** Support for CloudSec

- d. Support for ACI Multi-Pod
- 9. True or false: The Nexus 93180YC-FX leaf switch supports MACsec.
 - a. True
 - b. False
- 10. Which of the following platforms is a low-cost option for server CIMC and other low-bandwidth functions that rely on RJ-45 connectivity?
 - a. Nexus 9336C-FX2
 - b. Nexus 93180YC-FX
 - c. Nexus 9332C
 - d. Nexus 9348GC-FXP

Foundation Topics

ACI Topologies and Components

Like many other current data center fabrics, ACI fabrics conform to a Clos-based leaf-and-spine topology.

In ACI, leaf and spine switches are each responsible for different functions. Together, they create an architecture that is highly standardized across deployments. Cisco has introduced several new connectivity models and extensions for ACI fabrics over the years, but none of these changes break the core ACI topology that has been the standard from day one. Any topology modifications introduced in this section should therefore be seen as slight enhancements that help address specific use cases and not as deviations from the standard ACI topology.

Clos Topology

In his 1952 paper titled “A Study of Non-blocking Switching Networks,” Bell Laboratories researcher Charles Clos formalized how multistage telephone switching systems could be built to forward traffic, regardless of the number of calls served by the overall system.

The mathematical principles proposed by Clos also help address the challenge of needing to build highly scalable data centers using relatively low-cost switches.

[Figure 2-1](#) illustrates a three-stage Clos fabric consisting of one layer for ingress traffic, one layer for egress traffic, and a central layer for forwarding traffic between the layers. Multistage designs such as this can result in networks that are not oversubscribed or that are very close to not being oversubscribed.

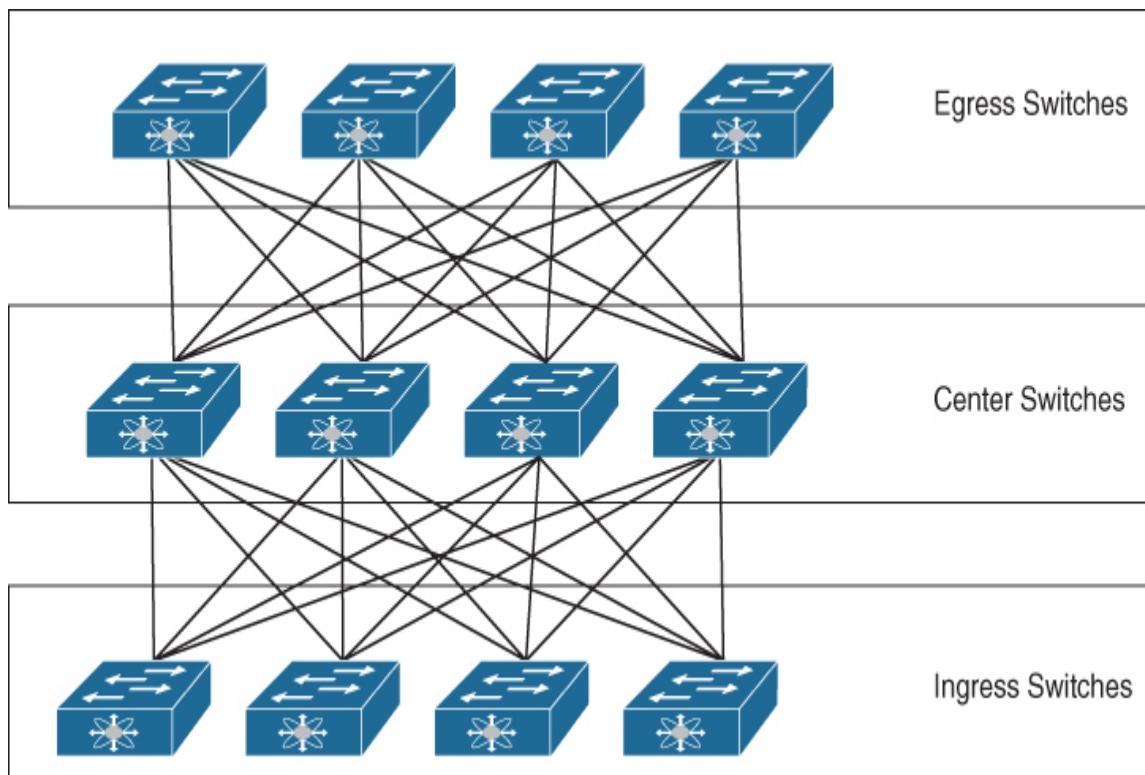


Figure 2-1 Conceptual View of a Three-Stage Clos Topology

Modern data center switches forward traffic at full duplex. Therefore, there is little reason to depict separate layers for ingress and egress traffic. It is possible to fold the top layer from the three-tier Clos topology in [Figure 2-1](#) into the bottom layer to achieve what the industry refers to as a “folded” Clos topology, illustrated in [Figure 2-2](#).

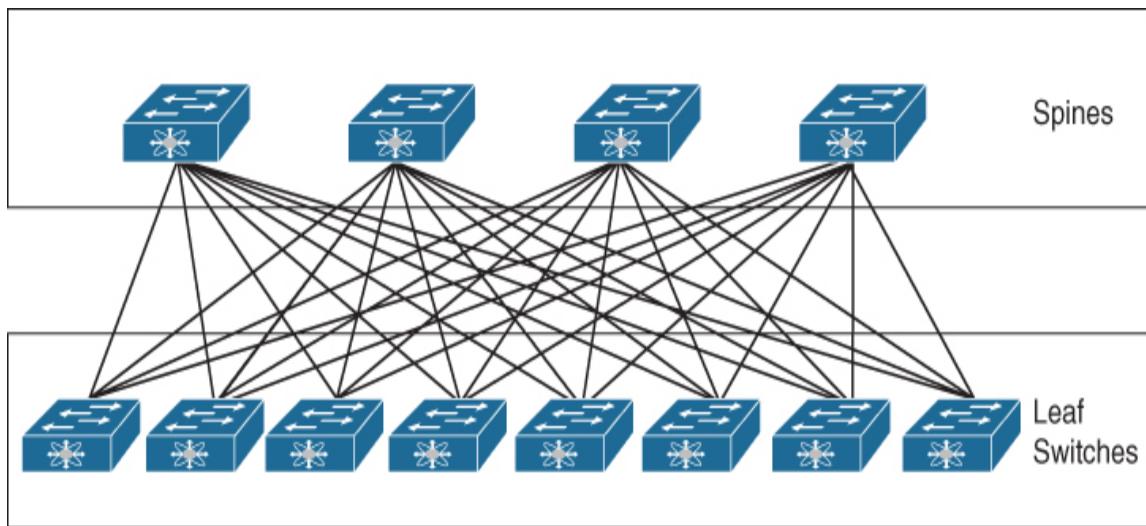


Figure 2-2 Folded Clos Topology

As indicated in [Figure 2-2](#), a leaf switch is an ingress/egress switch. A spine switch is an intermediary switch whose most critical function is to perform rapid forwarding of traffic between leaf switches. Leaf switches connect to spine switches in a full-mesh topology.

Note

At first glance, a three-tier Clos topology may appear to be similar to the traditional three-tier data center architecture. However, there are some subtle differences. First, there are no physical links between leaf switches in the Clos topology. Second, there are no

physical links between spine switches. The elimination of cross-links within each layer simplifies network design and reduces control plane complexity.

Standard ACI Topology

An ACI fabric forms a Clos-based spine-and-leaf topology and is usually depicted using two rows of switches. Depending on the oversubscription and overall network throughput requirements, the number of spines and leaf switches will be different in each ACI fabric.

Note

In the context of the Implementing Cisco Application Centric Infrastructure DCACI 300-620 exam, it does not matter whether you look at a given ACI fabric as a two-tiered Clos topology or as a three-tiered folded Clos topology. It is common for the standard ACI topology to be referred to as a two-tier spine-and-leaf topology.

[Figure 2-3](#) shows the required components and cabling for an ACI fabric. Inheriting from its Clos roots, no cables should be connected between ACI leaf switches. Likewise, ACI spines being cross-cabled results in ACI disabling the cross-connected ports. While the topology shows a full mesh of cabling between the spine-and-leaf layers, a fabric can operate without a full mesh. However, a full mesh of cables between layers is still recommended.

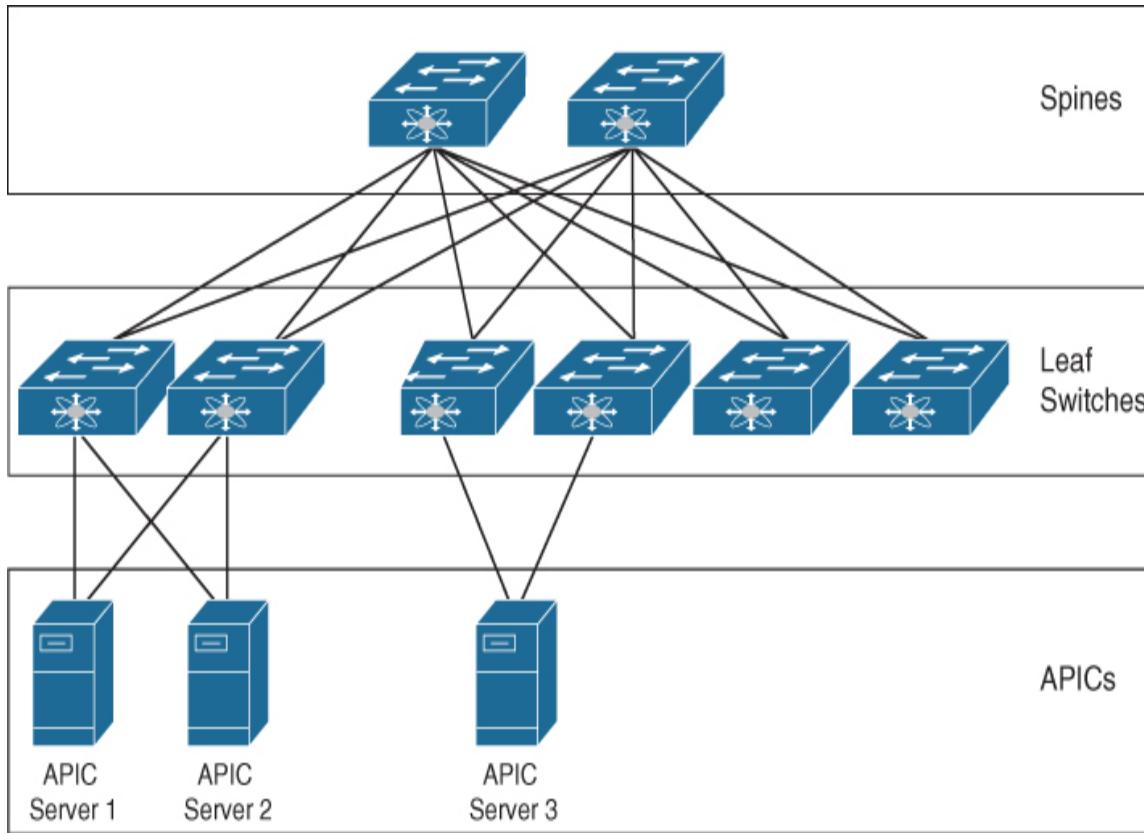


Figure 2-3 Standard ACI Fabric Topology

In addition to optics and cabling, the primary hardware components required to build an ACI fabric are as follows:

- **Application Policy Infrastructure Controllers (APICs):** The APICs are the brains of an ACI fabric and serve as the single source of truth for configuration within the fabric. A clustered set of (typically three) controllers attaches directly to leaf switches and provides management, policy programming, application deployment, and health monitoring for an ACI fabric. Note in [Figure 2-3](#) that APICs are not in the data path or the forwarding topology. Therefore, the failure of one or more APICs does not halt packet forwarding. An ACI fabric requires a minimum of one APIC, but an ACI fabric with one APIC should be used only for lab purposes.



- **Spine switches:** ACI spine switches are Clos intermediary switches that have a number of key functions. They exchange routing updates with leaf switches via Intermediate System-to-Intermediate System (IS-IS) and perform rapid forwarding of packets between leaf switches. They provide endpoint lookup services to leaf switches through the Council of Oracle Protocol (COOP). They also handle route reflection to leaf switches using Multiprotocol BGP (MP-BGP), allowing external routes to be distributed across the fabric regardless of the number of tenants. (All three of these are control plane protocols and are covered in more detail in future chapters.) Spine switches also serve as roots for multicast trees within a fabric. By default, all spine switch interfaces besides the mgmt0 port are configured as fabric ports. **Fabric ports** are the interfaces that are used to interconnect spine and leaf switches within a fabric.
- **Leaf switches:** Leaf switches are the ingress/egress points for traffic into and out of an ACI fabric. As such, they are the connectivity points for endpoints, including servers and appliances, into the fabric. Layer 2 and 3 connectivity from the outside world into an ACI fabric is also typically established via leaf switches. ACI security policy enforcement occurs on leaf switches. Each leaf switch has a number of high-bandwidth uplink ports preconfigured as fabric ports.

In addition to the components mentioned previously, optional hardware components that can be deployed alongside an ACI fabric include fabric extenders (FEX). Use of FEX solutions in ACI is not ideal because leaf hardware

models currently on the market are generally low cost and feature heavy compared to FEX technology.

FEX attachment to ACI is still supported to allow for migration of brownfield gear into ACI fabrics. The DCACI 300-620 exam does not cover specific FEX model support, so neither does this book.

Note

There are ways to extend an ACI fabric into a virtualized environment by using ACI Virtual Edge (AVE) and Application Virtual Switch (AVS). These are software rather than hardware components and are beyond the scope of the DCACI 300-620 exam.

Engineers may sometimes dedicate two or more leaf switches to a particular function. Engineers typically evaluate the following categories of leaf switches as potential options for dedicating hardware:

- **Border Leaf:** *Border leaf* switches provide Layer 2 and 3 connectivity between an ACI fabric and the outside world. Border leaf switches are sometimes points of policy enforcement between internal and external endpoints.



- **Service Leaf:** *Service leaf* switches are leaf switches that connect to Layer 4-7 service appliances, such as firewalls and load balancers.
- **Compute Leaf:** *Compute leaf* switches are ACI leaf switches that connect to servers. Compute leaf

switches are points of policy enforcement when traffic is being sent between local endpoints.

- **IP Storage Leaf:** *IP storage leaf* switches are ACI leaf switches that connect to IP storage systems. IP storage leaf switches can also be points of policy enforcement for traffic to and from local endpoints.

There are scalability benefits associated with dedicating leaf switches to particular functions, but if the size of the network does not justify dedicating leaf switches to a function, consider at least dedicating a pair of leaf switches as border leaf switches. Service leaf functionality can optionally be combined with border leaf functionality, resulting in the deployment of a pair (or more) of collapsed border/service leaf switches in smaller environments.

Cisco publishes a Verified Scalability Guide for each ACI code release. At the time of this writing, 500 is considered the maximum number of leaf switches that can be safely deployed in a single fabric that runs on the latest code.

ACI Stretched Fabric Topology

A *stretched ACI fabric* is a partially meshed design that connects ACI leaf and spine switches distributed in multiple locations. The stretched ACI fabric design helps lower deployment costs when full-mesh cable runs between all leaf and spine switches in a fabric tend to be cost-prohibitive.

[Figure 2-4](#) shows a stretched ACI fabric across two sites.

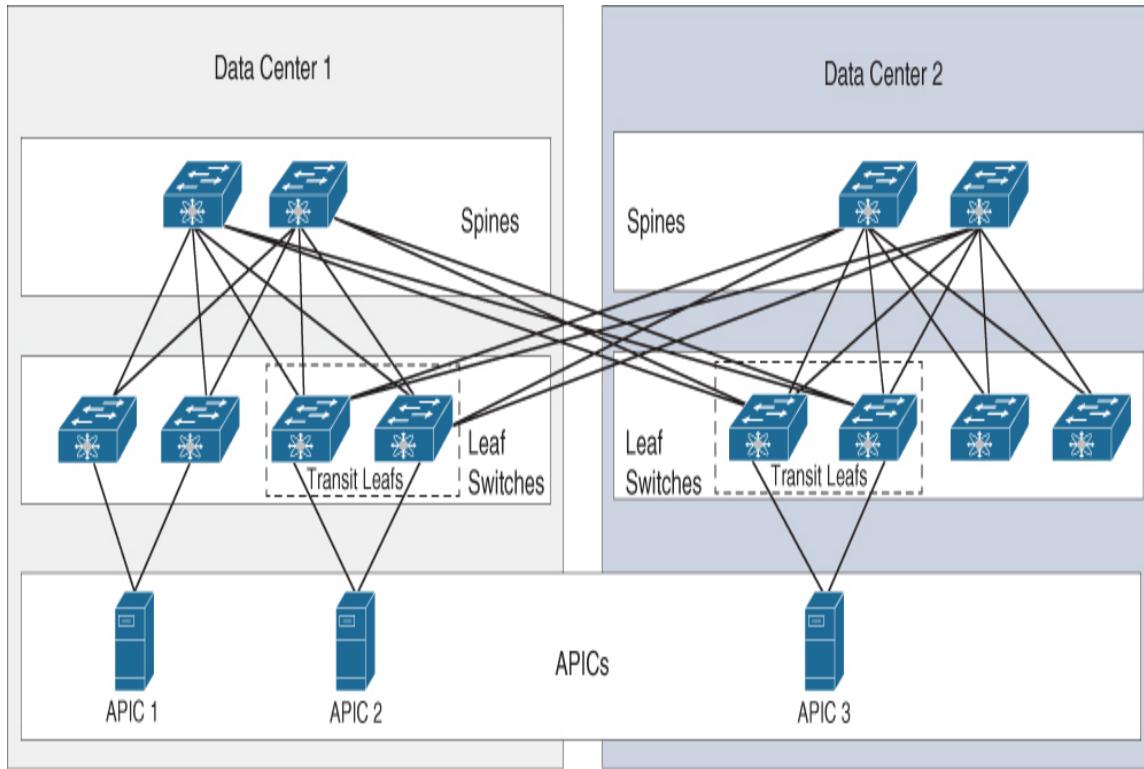


Figure 2-4 ACI Stretched Fabric Topology

A stretched fabric amounts to a single administrative domain and a single availability zone. Because APICs in a stretched fabric design tend to be spread across sites, cross-site latency is an important consideration. APIC clustering has been validated across distances of 800 kilometers between two sites.

A new term introduced in Figure 2-4 is *transit leaf*. A **transit leaf** is a leaf switch that provides connectivity between two sites in a stretched fabric design. Transit leaf switches connect to spine switches in both sites. No special configuration is required for transit leaf switches. At least one transit leaf switch must be provisioned in each site for redundancy reasons.

While stretched fabrics simplify extension of an ACI fabric, this design does not provide the benefits of newer topologies such as ACI Multi-Pod and ACI Multi-Site and

stretched fabrics are therefore no longer commonly deployed or recommended.

ACI Multi-Pod Topology



The **ACI Multi-Pod** topology is a natural evolution of the ACI stretched fabric design in which spine and leaf switches are divided into pods, and different instances of IS-IS, COOP, and MP-BGP protocols run inside each pod to enable a level of control plane fault isolation.

Spine switches in each pod connect to an interpod network (IPN). Pods communicate with one another through the IPN. [Figure 2-5](#) depicts an ACI Multi-Pod topology.



An ACI Multi-Pod IPN has certain requirements that include support for OSPF, end-to-end IP reachability, DHCP relay capabilities on the last-hop routers that connect to spines in each pod, and an increased maximum transmission unit (MTU). In addition, a Multi-Pod IPN needs to support forwarding of multicast traffic (PIM-Bidir) to allow the replication of broadcast, unknown unicast, and multicast (BUM) traffic across pods.

One of the most significant use cases for ACI Multi-Pod is active/active data center design. Although ACI Multi-Pod supports a maximum round-trip time latency of 50 milliseconds between pods, most Multi-Pod deployments are

often built to achieve active/active functionality and therefore tend to have latencies of less than 5 milliseconds.

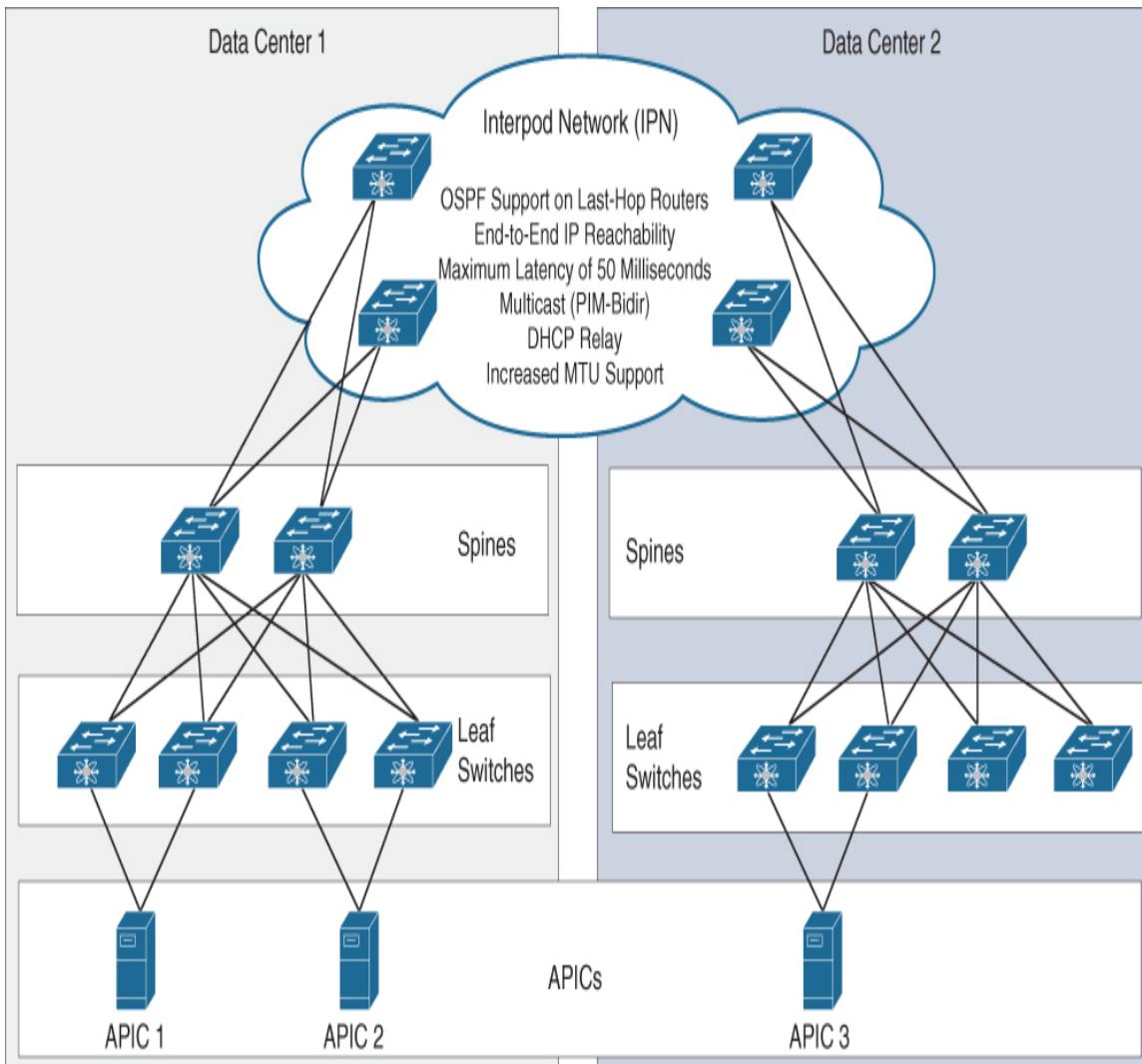


Figure 2-5 ACI Multi-Pod Topology

Note

Another solution that falls under the umbrella of ACI Multi-Pod is Virtual Pod (vPod). ACI vPod is not a new topology per se. It is an extension of a Multi-Pod fabric in the form of a new pod at a remote location where at least two ESXi servers are available, and deployment of ACI hardware is not desirable. ACI vPod components

needed at the remote site for this solution include virtual spine (vSpine) appliances, virtual leaf (vLeaf) appliances, and the Cisco ACI Virtual Edge. ACI vPod still requires a physical ACI footprint since vPod is managed by the overall Multi-Pod APIC cluster.

On the issue of scalability, it should be noted that as of the time of writing, 500 is the maximum number of leaf switches that can be safely deployed within a single ACI fabric. However, the Verified Scalability Guide for the latest code revisions specifies 400 as the absolute maximum number of leaf switches that can be safely deployed in each pod. Therefore, for a fabric to reach its maximum supported scale, leaf switches should be deployed across at least 2 pods within a Multi-Pod fabric. Each pod supports deployment of 6 spines, and each Multi-Pod fabric currently supports the deployment of up to 12 pods.

[Chapter 16, “ACI Anywhere,”](#) covers ACI Multi-Pod in more detail. For now, understand that Multi-Pod is functionally a single fabric and a single availability zone, even though it does not represent a single network failure domain.

ACI Multi-Site Topology



ACI Multi-Site is a solution that interconnects multiple ACI fabrics for the purpose of homogenous policy deployment across ACI fabrics, homogenous security policy deployment across on-premises ACI fabrics and public clouds, and cross-site stretched subnet capabilities, among others.

**Key
Topic**

In an ACI Multi-Site design, each ACI fabric has its own dedicated APIC cluster. A clustered set of three nodes called Multi-Site Orchestrator (MSO) establishes API calls to each fabric independently and can configure tenants within each fabric with desired policies.

Note

Nodes forming an MSO cluster have traditionally been deployed as VMware ESXi virtual machines (VMs). Cisco has recently introduced the ability to deploy an MSO cluster as a distributed application (.aci format) on Cisco Application Services Engine (ASE). Cisco ASE is a container-based solution that provides a common platform for deploying and managing Cisco data center applications. ASE can be deployed in three form factors: a physical form factor consisting of bare-metal servers, a virtual machine form factor for on-premises deployments via ESXi or Linux KVM hypervisors, and a virtual machine form factor deployable within a specific Amazon Web Services (AWS) region.

[Figure 2-6](#) shows an ACI Multi-Site topology that leverages a traditional VM-based MSO cluster.

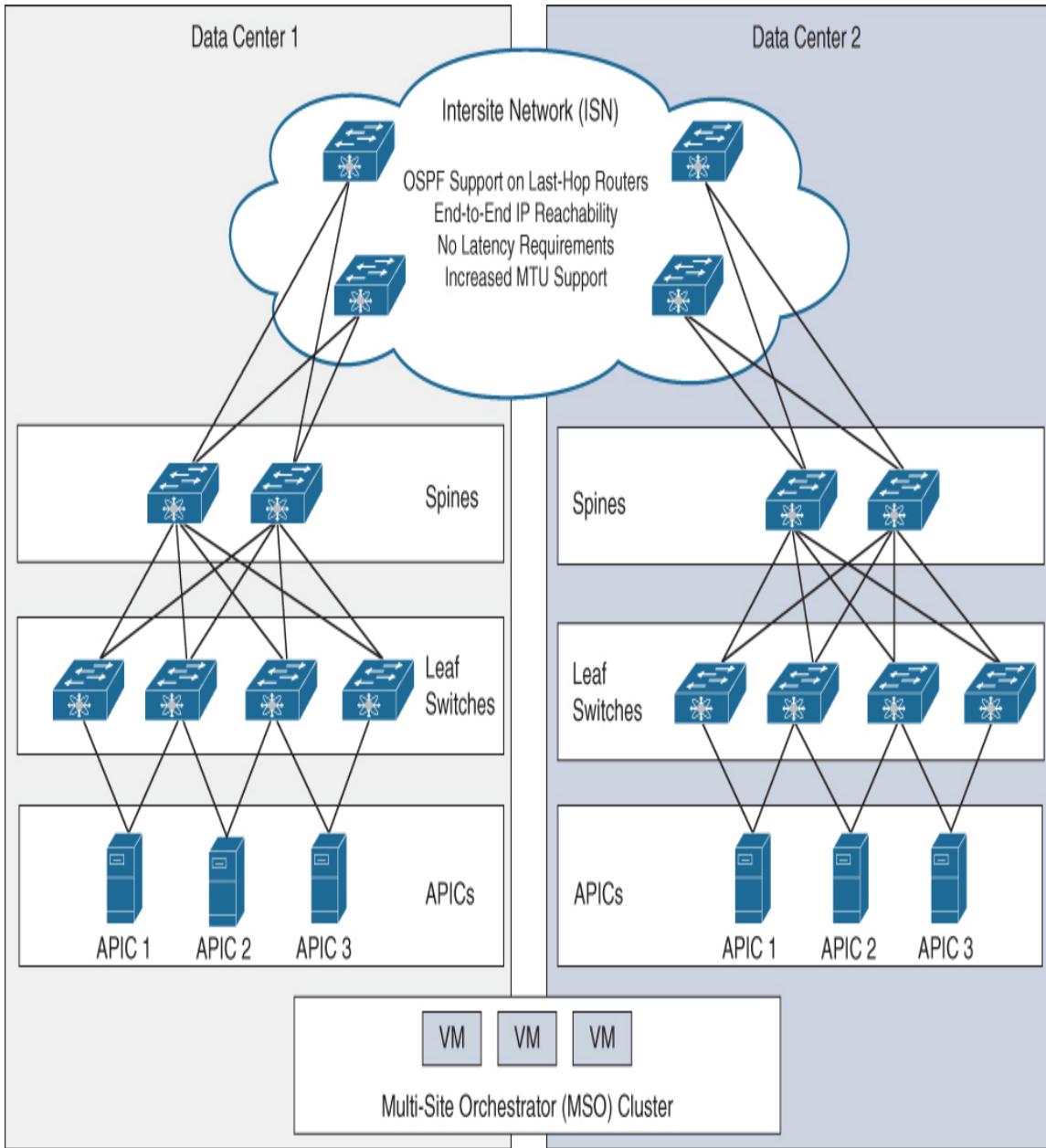


Figure 2-6 ACI Multi-Site Topology

Key Topic

As indicated in [Figure 2-6](#), end-to-end communication between sites in an ACI Multi-Site design requires the use of an intersite network (ISN). An ACI Multi-Site ISN faces less

stringent requirements compared to ACI Multi-Pod IPNs. In an ISN, end-to-end IP connectivity between spines across sites, OSPF on the last-hop routers connecting to the spines, and increased MTU support allowing VXLAN-in-IP encapsulation are all still required. However, ACI Multi-Site does not dictate any cross-site latency requirements, nor does it require support for multicast or DHCP relay within the ISN.

ACI Multi-Site does not impose multicast requirements on the ISN because ACI Multi-Site has been designed to accommodate larger-scale ACI deployments that may span the globe. It is not always feasible or expected for a company that has a global data center footprint to also have a multicast backbone spanning the globe and between all data centers.



Due to the introduction of new functionalities that were not required in earlier ACI fabrics, Cisco introduced a second generation of spine hardware. Each ACI fabric within an ACI Multi-Site design requires at least one second-generation or newer piece of spine hardware for the following reasons:

- **Ingress replication of BUM traffic:** To accommodate BUM traffic forwarding between ACI fabrics without the need to support multicast in the ISN, Multi-Site-enabled spines perform ingress replication of BUM traffic. This function is supported only on second-generation spine hardware.
- **Cross-fabric namespace normalization:** Each ACI fabric has an independent APIC cluster and therefore an independent brain. When policies and parameters are communicated between fabrics in VXLAN header

information, spines receiving cross-site traffic need to have a way to swap remotely significant parameters, such as VXLAN network identifiers (VNIDs), with equivalent values for the local site. This function, which is handled in hardware and is called *namespace normalization*, requires second-generation or newer spines.

Note that in contrast to ACI Multi-Site, ACI Multi-Pod *can* be deployed using first-generation spine switches.

For ACI Multi-Site deployments, current verified scalability limits published by Cisco suggest that fabrics with stretched policy requirements that have up to 200 leaf switches can be safely incorporated into ACI Multi-Site. A single ACI Multi-Site deployment can incorporate up to 12 fabrics as long as the total number of leaf switches in the deployment does not surpass 1600.

Each fabric in an ACI Multi-Site design forms a separate network failure domain and a separate availability zone.

ACI Multi-Tier Architecture

Introduced in Release 4.1, ACI Multi-Tier provides the capability for vertical expansion of an ACI fabric by adding an extra layer or tier of leaf switches below the standard ACI leaf layer.

With the Multi-Tier enhancement, the standard ACI leaf layer can also be termed the Tier 1 leaf layer. The new layer of leaf switches that are added to vertically expand the fabric is called the Tier 2 leaf layer. [Figure 2-7](#) shows these tiers. APICs, as indicated, can attach to either Tier 1 or Tier 2 leaf switches.

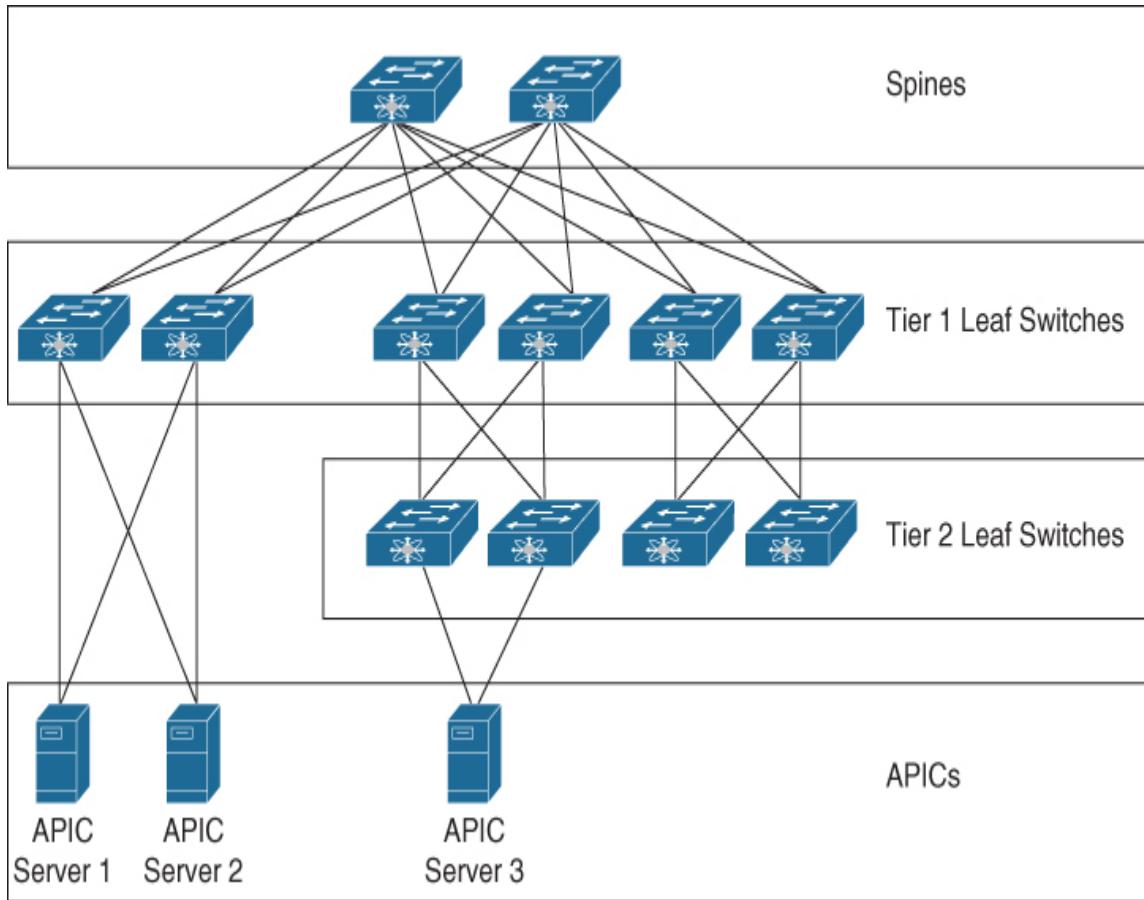


Figure 2-7 ACI Multi-Tier Topology

Note

The topology shown in [Figure 2-7](#) goes against the requirement outlined earlier in this chapter, in the section “Standard ACI Topology,” *not* to cross-connect leaf switches. The ACI Multi-Tier architecture is an exception to this rule. Leaf switches within each tier, however, still should never be cross-connected.

An example of a use case for ACI Multi-Tier is the extension of an ACI fabric across data center halls or across buildings that are in relatively close proximity while minimizing long-distance cabling and optics requirements. Examine the diagram in [Figure 2-8](#). Suppose that an enterprise data

center has workloads in an alternate building. In this case, the company can deploy a pair of Tier 1 leaf switches in the new building and expand the ACI fabric to the extent needed within the building by using a Tier 2 leaf layer. Assuming that 6 leaf switches would have been required to accommodate the port requirements in the building, as [Figure 2-8](#) suggests, directly cabling these 6 leaf switches to the spines as Tier 1 leaf switches would have necessitated 12 cross-building cables. However, the use of an ACI Multi-Tier design enables the deployment of the same number of switches using 4 long-distance cable runs.

ACI Multi-Tier can also be an effective solution for use within data centers in which the cable management strategy is to minimize inter-row cabling and relatively low-bandwidth requirements exist for top-of-rack switches. In such a scenario, Tier 1 leaf switches can be deployed end-of-row, and Tier 2 leaf switches can be deployed top-of-rack.

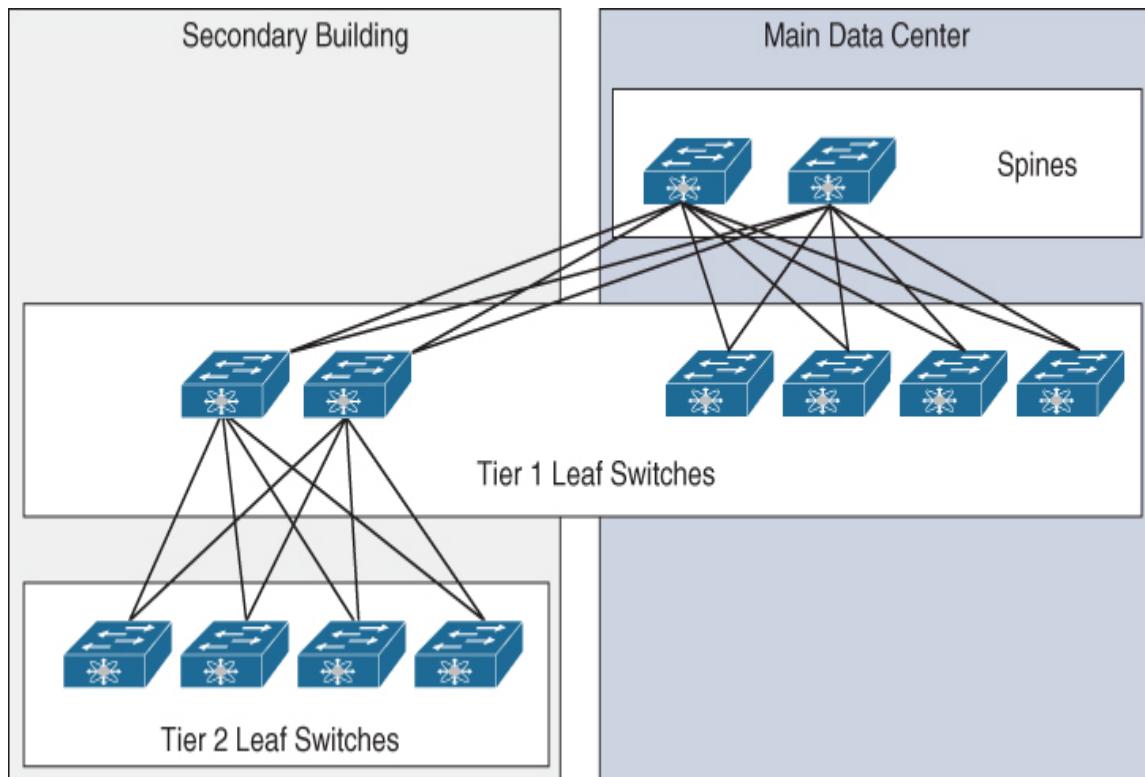


Figure 2-8 Extending an ACI Fabric by Using ACI Multi-Tier in an Alternative Location

Note

ACI Multi-Tier *might not* be a suitable solution if the amount of bandwidth flowing upstream from Tier 2 leaf switches justifies the use of dedicated uplinks to spines.

Not all ACI switch platforms support Multi-Tier functionality.

Remote Leaf Topology



For remote sites in which data center endpoints may be deployed but their number and significance do not justify the deployment of an entirely new fabric or pod, the ACI *Remote Leaf* solution can be used to extend connectivity and ensure consistent policies between the main data center and the remote site. With such a solution, leaf switches housed at the remote site communicate with spines and APICs at the main data center over a generic IPN. Each Remote Leaf switch can be bound to a single pod.

There are three main use cases for Remote Leaf deployments:

- **Satellite/small colo data centers:** If a company has a small data center consisting of several top-of-rack switches and the data center may already have dependencies on a main data center, this satellite data center can be integrated into the main data center by using the Remote Leaf solution.

- **Data center extension and migrations:** Cross-data center migrations that have traditionally been done through Layer 2 extension can instead be performed by deploying a pair of Remote Leafs in the legacy data center. This approach often has cost benefits compared to alternative Layer 2 extension solutions if there is already an ACI fabric in the target state data center.
- **Telco 5G distributed data centers:** Telcom operators that are transitioning to more distributed mini data centers to bring services closer to customers but still desire centralized management and consistent policy deployment across sites can leverage Remote Leaf for these mini data centers.

In addition to these three main use cases, disaster recovery (DR) is sometimes considered a use case for Remote Leaf deployments, even though DR is a use case more closely aligned with ACI Multi-Site designs.

In a Remote Leaf solution, the APICs at the main data center deploy policy to the Remote Leaf switches as if they were locally connected.

[Figure 2-9](#) illustrates a Remote Leaf solution.

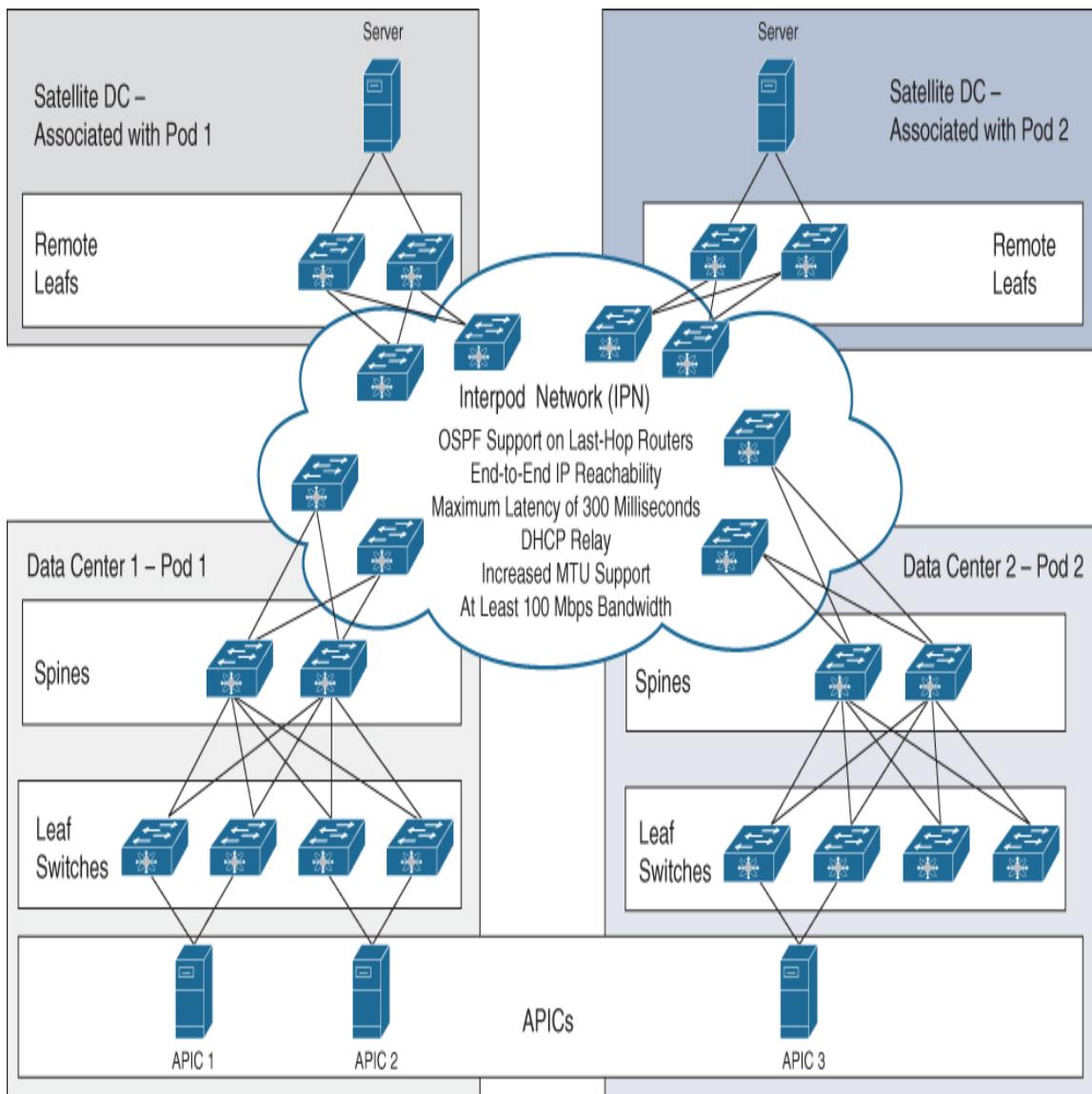


Figure 2-9 Remote Leaf Topology and IPN Requirements

IPN requirements for a Remote Leaf solution are as follows:

- **MTU:** The solution must support an end-to-end MTU that is at least 100 bytes higher than that of the endpoint source traffic. Assuming that 1500 bytes has been configured for data plane MTU, Remote Leaf can be deployed using a minimum MTU of 1600 bytes. An IPN MTU this low, however, necessitates that ACI

administrators lower the ACI fabricwide control plane MTU, which is 9000 bytes by default.

- **Latency:** Up to 300 milliseconds latency between the main data center and remote location is acceptable.
- **Bandwidth:** Remote Leaf is supported with a minimum IPN bandwidth of 100 Mbps.
- **VTEP reachability:** A Remote Leaf switch logically associates with a single pod if integrated into a Multi-Pod solution. To make this association possible, the Remote Leaf should be able to route traffic over the IPN to the VTEP pool of the associated pod. Use of a dedicated VRF for IPN traffic is recommended where feasible.
- **APIC infra IP reachability:** A Remote Leaf switch needs IP connectivity with all APICs in a Multi-Pod cluster at the main data center. If an APIC has assigned itself IP addresses from a VTEP range different than the pod VTEP pool, the additional VTEP addresses need to also be advertised over the IPN.
- **OSPF support on upstream routers:** Routers northbound of both the Remote Leaf switches and the spine switches need to support OSPF and must be able to encapsulate traffic destined to directly attached ACI switches using VLAN 4. This requirement exists only for directly connected devices and does not extend end-to-end in the IPN.
- **DHCP relay:** The upstream router directly connected to Remote Leaf switches needs to enable DHCP relay to relay DHCP packets to the APIC IP addresses in the infra tenant. The DHCP relay configuration needs to be applied on the VLAN 4 subinterface or SVI.

Note that unlike a Multi-Pod IPN, a Remote Leaf IPN does not require Multicast PIM-Bidir support. This is because the Remote Leaf solution uses headend replication (HER) tunnels to forward BUM traffic between sites.

In a Remote Leaf design, traffic between known local endpoints at the remote site is switched directly, whether physically or virtually. Any traffic whose destination is in ACI but is unknown or not local to the remote site is forwarded to the main data center spines.

Note

[Chapter 16](#) details MTU requirements for IPN and ISN environments for ACI Multi-Pod and ACI Multi-Site. It also covers how to lower control plane and data plane MTU values within ACI if the IPN or ISN does not support high MTU values. Although it does not cover Remote Leaf, the same general IPN MTU concepts apply.

Not all ACI switches support Remote Leaf functionality. The current maximum verified scalability number for Remote Leaf switches is 100 per fabric.

APIC Clusters

The ultimate size of an APIC cluster should be directly proportionate to the size of the Cisco ACI deployment. From a management perspective, any active APIC controller in a cluster can service any user for any operation. Controllers can be transparently added to or removed from a cluster.



APICs can be purchased either as physical or virtual appliances. Physical APICs are 1 rack unit (RU) Cisco C-Series servers with ACI code installed and come in two different sizes: M for medium and L for large. In the context of APICs, “size” refers to the scale of the fabric and the number of endpoints. Virtual APICs are used in ACI mini deployments, which consist of fabrics with up to two spine switches and four leaf switches.



As hardware improves, Cisco releases new generations of APICs with updated specifications. At the time of this writing, Cisco has released three generations of APICs. The first generation of APICs (M1/L1) shipped as Cisco UCS C220 M3 servers. Second-generation APICs (M2/L2) were Cisco UCS C220 M4 servers. Third-generation APICs (M3/L3) are shipping as UCS C220 M5 servers.

[Table 2-2](#) details specifications for current M3 and L3 APICs.

Table 2-2 M3 and L3 APIC Specifications

Com pon ent	M3	L3
Processor	2x 1.7 GHz Xeon scalable 3106/85W 8C/11MB cache/DDR4 2133MHz	2x 2.1 GHz Xeon scalable 4110/85W 8C/11MB cache/DDR4 2400MHz

Memory	6x 16 GB DDR4-2666-MHz RDIMM/PC4-21300/single rank/x4/1.2v	12x 16 GB DDR4-2666-MHz RDIMM/PC4-21300/single rank/x4/1.2v
Hard drive	2x 1 TB 12G SAS 7.2K RPM SFF HDD	2x 2.4 TB 12G SAS 10K RPM SFF HDD (4K)
Network cards	1x Cisco UCS VIC 1455 Quad Port 10/25G SFP28 CNA PCIE	1x Cisco UCS VIC 1455 Quad Port 10/25G SFP28 CNA PCIE

Note in [Table 2-2](#) that the only differences between M3 and L3 APICs are the sizes of their CPUs, memory, and hard drives. This is because fabric growth necessitates that increased transaction rates be supported, which drives up compute requirements.

[Table 2-3](#) shows the hardware requirements for virtual APICs.

Table 2-3 Virtual APIC Specifications

Component	Virtual APIC
Processor	8 vCPUs
Memory	32 GB

Hard drive*	300 GB HDD 100 GB SSD
Supported ESXi hypervisor version	6.5 or above

* A VM is deployed with two HDDs.

APIC Cluster Scalability and Sizing

APIC cluster hardware is typically purchased from Cisco in the form of a bundle. An APIC bundle is a collection of one or more physical or virtual APICs, and the bundle that needs to be purchased depends on the desired target state scalability of the ACI fabric.

[Table 2-4](#) shows currently available APIC cluster hardware options and the general scalability each bundle can individually achieve.

Table 2-4 APIC Hardware Bundles

Part Number	Number of APICs	General Scalability
APIC-CLUSTER-XS (ACI mini bundle)	1 M3 APIC, 2 virtual APICs, and 2 Nexus 9332C spine switches	Up to 2 spines and 4 leaf switches

APIC-CLUSTER-M3	3 M3 APICs	Up to 1200 edge ports
APIC-CLUSTER-L3	3 L3 APICs	More than 1200 edge ports

APIC-CLUSTER-XS specifically addresses ACI mini fabrics. ACI mini is a fabric deployed using two Nexus 9332C spine switches and up to four leaf switches. ACI mini is suitable for lab deployments, small colocation deployments, and deployments that are not expected to span beyond four leaf switches.

APIC-CLUSTER-M3 is designed for medium-sized deployments where the number of server ports connecting to ACI is not expected to exceed 1200, which roughly translates to 24 leaf switches.

APIC-CLUSTER-L3 is a bundle designed for large-scale deployments where the number of server ports connecting to ACI exceeds or will eventually exceed 1200.

Beyond bundles, Cisco allows customers to purchase individual APICs for the purpose of expanding an APIC cluster to enable further scaling of a fabric. Once a fabric expands beyond 1200 edge ports, ACI Verified Scalability Guides should be referenced to determine the optimal number of APICs for the fabric.

According to Verified Scalability Guides for ACI Release 4.1(1), an APIC cluster of three L3 APICs should suffice in deployments with up to 80 leaf switches. However, the

cluster size would need to be expanded to four or more APICs to allow a fabric to scale up to 200 leaf switches.

Note

Cisco recommends against deployment of APIC cluster sizes of 4 and 6. Current recommended cluster sizes are 3, 5, or 7 APICs per fabric.

Each APIC cluster houses a distributed multi-active database in which processes are active on all nodes. Data, however, is distributed or sliced across APICs via a process called *database sharding*. **Sharding** is a result of the evolution of what is called horizontal partitioning of databases and involves distributing a database across multiple instances of the schema. Sharding increases both redundancy and performance because a large partitioned table can be split across multiple database servers. It also enables a scale-out model involving adding to the number of servers as opposed to having to constantly scale up servers through hardware upgrades.

ACI shards each attribute within the APIC database to three nodes. A single APIC out of the three is considered active (the leader) for a given attribute at all times. If the APIC that houses the active copy of a particular slice or partition of data fails, the APIC cluster is able to recover via the two backup copies of the data residing on the other APICs. This is why the deployment of a minimum of three APICs is advised. Any APIC cluster deployed with fewer than three APICs is deemed unsuitable for production uses. Note that only the APIC that has been elected leader for a given attribute can modify the attribute.

[Figure 2-10](#) provides a conceptual view of data sharding across a three-APIC cluster. For each data set or attribute

depicted, a single APIC is elected leader. Assume that the active copy indicates that the APIC holding the active copy is leader for the given attribute.

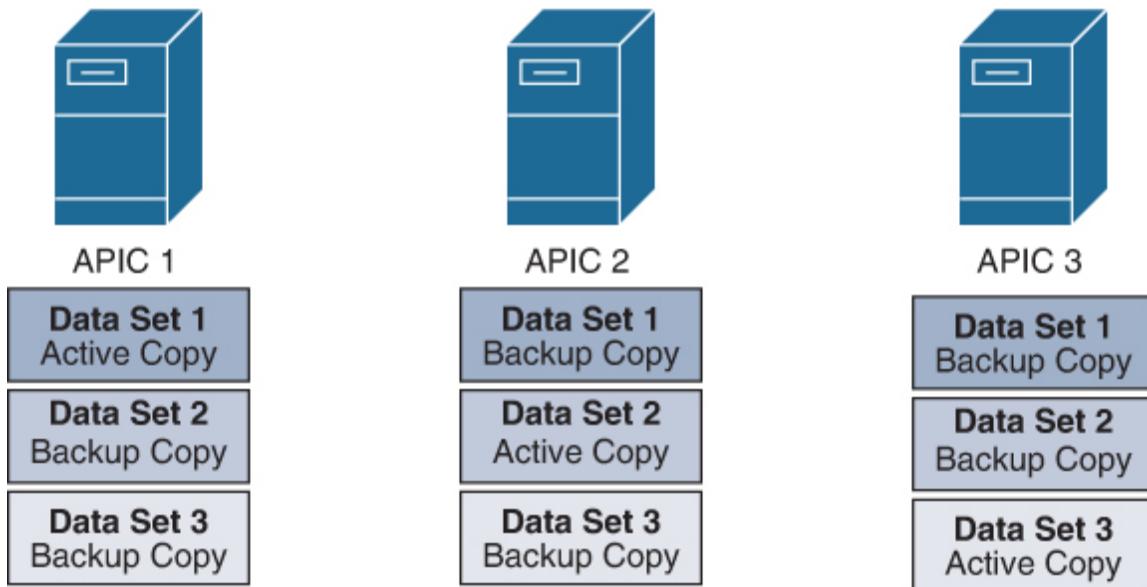


Figure 2-10 *Data Sharding Across Three APICs*

For a portion of a database to allow writes (configuration changes), a quorum of APICs housing the pertinent database attributes undergoing a write operation must be healthy and online. Because each attribute in an APIC database is sharded into three copies, a quorum is defined as two copies. If two nodes in a three-node APIC cluster were to fail simultaneously, the remaining APIC would move the entire database into a read-only state, and no configuration changes would be allowed until the quorum was restored.

When an APIC cluster scales to five or seven APICs, the sharding process remains unchanged. In other words, the number of shards of a particular subset of data does not increase past three, but the cluster further distributes the shards. This means that cluster expansion past three APICs

does not increase the redundancy of the overall APIC database.

[Figure 2-11](#) illustrates how an outage of Data Center 2, which results in the failure of two APICs, could result in portions of the APIC database moving into a read-only state. In this case, the operational APICs have at least two shards for Data Sets 1 and 3, so administrators can continue to make configuration changes involving these database attributes. However, Data Set 2 is now in read-only mode because two replicas of the attribute in question have been lost.

As [Figure 2-11](#) demonstrates, increasing APIC cluster size to five or seven does not necessarily increase the redundancy of the overall cluster.

A general recommendation in determining APIC cluster sizes is to deploy three APICs in fabrics scaling up to 80 leaf switches. If recoverability is a concern, a standby APIC can be added to the deployment. A total of five or seven APICs should be deployed for scalability purposes in fabrics expanding beyond 80 leaf switches.

If, for any reason, a fabric with more than three APICs is bifurcated, the APIC cluster attempts to recover this split-brain event. Once connectivity across all APICs is restored, automatic reconciliation takes place within the cluster, based on timestamps.

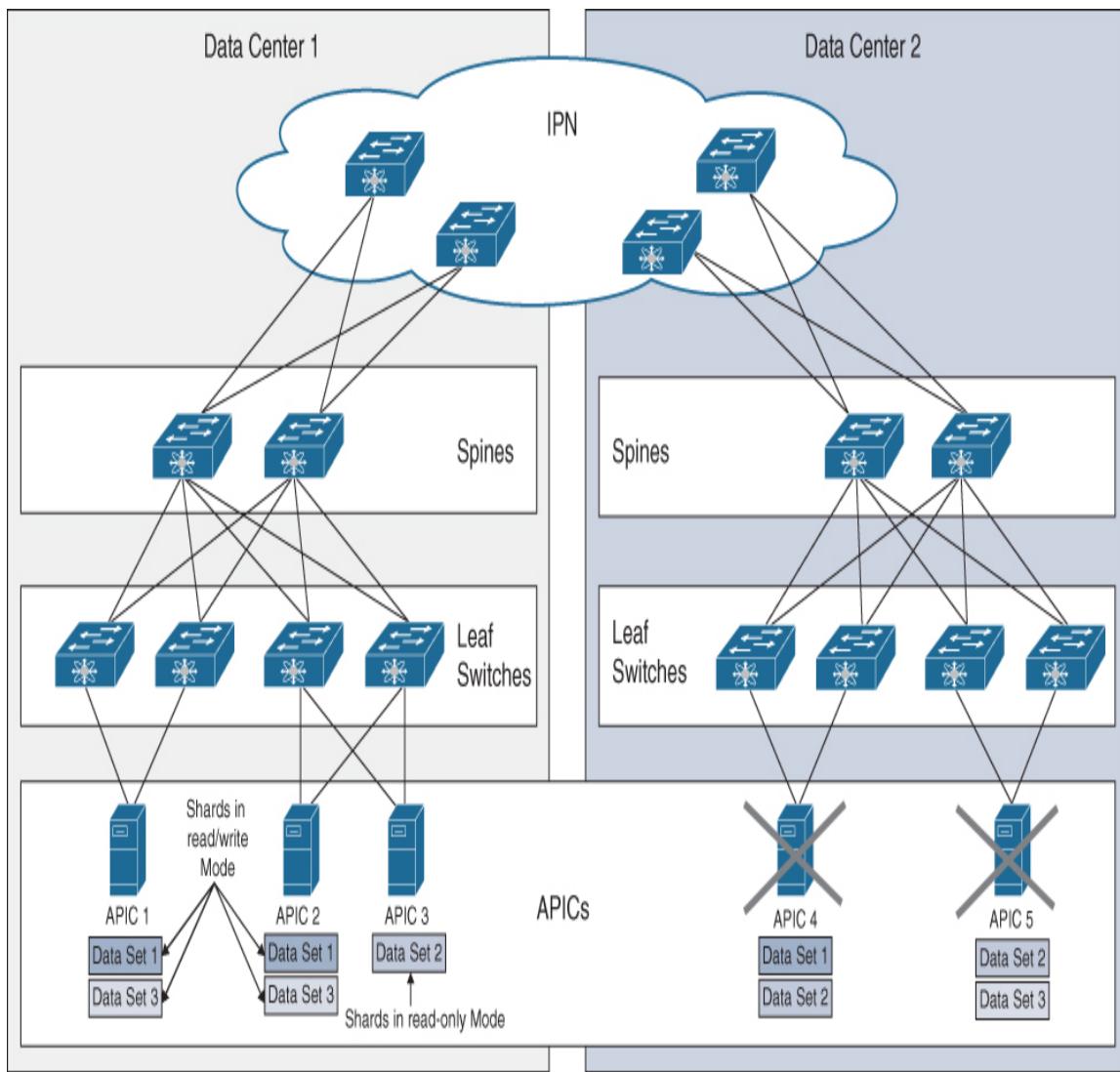


Figure 2-11 Impact of APIC Failures in a Five-Node Cluster

What would happen if Data Center 1 in [Figure 2-11](#) failed instead of Data Center 2, and all shards for a specific subset of data resided in Data Center 1 at the time of the outage? In such a scenario, the failure of three APICs could lead to the hypothetical loss of all three shards of a specific subset of data. To ensure that a total loss of a given pod does not result in the loss of all shards for a given attribute, Cisco recommends that no more than two APICs be placed in a single pod.

Note

Standby APICs allow an administrator to commission an APIC to allow recoverability of a fabric during failure scenarios in which the APIC quorum has been lost.

When a standby APIC is deployed in a fabric, it acts as a passive player. It does not actively service users or configure ACI switches. It also does not synchronize data with active APICs. When first deploying a controller as a standby APIC, at least three APICs in the cluster need to be active.

Spine Hardware

Cisco ACI spine hardware options includes Nexus 9300 Series fixed form factor switches as well as Nexus 9500 modular switches. Not all switches in the noted switch families can be deployed in ACI mode.

The primary factors that guide spine purchasing decisions are desired port bandwidths, feature requirements, hardware generation, and the required number of target state ports.

Whereas a fixed spine switch has a limited number of ports, a port in a modular platform can scale with the addition of more line cards to a chassis. For this reason, modular chassis are more suitable for fabrics that require massive scale.

Fixed spine platforms satisfy the scalability requirements of small to medium fabrics without problem.

First-Generation Spine Switches

As noted earlier in this chapter, first-generation spine switches are not supported as spines interconnecting ACI fabrics in ACI Multi-Site deployments. Other new solutions, such as Remote Leaf and ACI Multi-Tier also require second-generation spine switches. Understanding first-generation spine platforms is, however, beneficial for historical purposes because a large number of ACI deployments still contain first-generation hardware.

First-generation ACI spine switch models on the market at the time of this writing have model numbers that end in PQ. [Table 2-5](#) lists first-generation Nexus spine switches.



Table 2-5 First-Generation Spine Switches

Characteristic	Nexus 9336PQ	Nexus 9736PQ
Form factor	2 RU fixed switch	Line card for modular chassis
Supported modular platforms	N/A	Nexus 9504 Nexus 9508 Nexus 9516

40 Gigabit Ethernet ports	36 ports	36 ports
100 Gigabit Ethernet ports	N/A	N/A
ACI Multi-Pod support	Yes	Yes
CloudSec support	No	No
Remote Leaf support	No	No
ACI Multi-Tier support	No	No
ACI Multi-Site support	No	No

Even though first-generation spine switches do not support namespace normalization or ingress replication of BUM traffic, they can coexist with second-generation spine switches within a fabric. This coexistence enables companies to integrate fabrics into ACI Multi-Site without having to decommission older spines before the regular hardware refresh cycle.

Note

First-generation spine switches can no longer be ordered from Cisco.

Second-Generation Spine Switches

In addition to providing support for ACI Multi-Site, Remote Leaf, and ACI Multi-Tier, second-generation spine switch ports operate at both 40 Gigabit Ethernet and 100 Gigabit Ethernet speeds and therefore enable dramatic fabric bandwidth upgrades.

Second-generation spine switches also support MACsec and CloudSec. MACsec enables port-to-port encryption of traffic in transit at line rate. CloudSec enables cross-site encryption at line rate, eliminating the need for intermediary devices to support or perform encryption. Cross-site encryption is also referred to as *VTEP-to-VTEP encryption*.

Second-generation ACI spine switch models on the market at the time of this writing have model numbers that end in C, EX, and FX. [Table 2-6](#) provides additional details about second-generation spine platforms.

Key Topic

Table 2-6 Second-Generation Spine Switches

Characteristic	Nexus 9364C	Nexus 9332C	Nexus 9732C-EX	Nexus 9736C-FX
Form factor	2 RU fixed	1 RU fixed	Line card for modular	Line card for modular

			chassis	chassis
Supported modular platforms	N/A	N/A	Nexus 9504 Nexus 9508 Nexus 9516	Nexus 9504 Nexus 9508 Nexus 9516
40/100 Gigabit Ethernet ports	64	32	32	36
ACI Multi-Pod support	Yes	Yes	Yes	Yes
CloudSec support	Last 16 ports	Last 8 ports	N/A	All ports
Remote Leaf support	Yes	Yes	Yes	Yes
ACI Multi-Tier support	Yes	Yes	Yes	Yes
	Yes	Yes	Yes	Yes

ACI Multi-Site support			
------------------------	--	--	--

In addition to the hardware listed in [Table 2-6](#), Nexus 9732C-FX line cards will be supported as ACI spine line cards in the near future.

New spine switches with 100/400 Gigabit Ethernet ports are also on the horizon. The Nexus 9316D-GX is already available and is supported as an ACI spine. This platform is also in the roadmap for support as a leaf switch. The 100/400 Gigabit Ethernet Nexus 93600CD-GX switch, which is supported as an ACI leaf, is also in the roadmap for use as a spine.

Cisco uses the term *cloud scale* to refer to the newer Nexus switch models that contain the specialized ASICs needed for larger buffer sizes, larger endpoint tables, and visibility into packets and flows traversing the switch without impacting CPU utilization. Second-generation ACI spine switches fall into the category of cloud-scale switches.

Leaf Hardware

Cisco ACI leaf hardware options include Nexus 9300 Series fixed form factor switches. Not all switches in the noted switch families can be deployed in ACI mode.

The primary factors that guide leaf purchasing decisions are the desired port bandwidths, feature requirements, hardware generation, and the required number of target state ports.

First-Generation Leaf Switches

First-generation ACI leaf switches are Nexus 9300 Series platforms that are based on the Application Leaf Engine (ALE) ASICs.

The hardware resources that enable whitelisting of traffic are ternary content-addressable memory (TCAM) resources, referred to as the *policy CAM*.

Policy CAM sizes vary depending on the hardware. The policy CAM size and behavior limitations in first-generation switches tended to sometimes limit whitelisting projects.

There are also a number of other capability differences between first- and second-generation leaf hardware, such as handling of Layer 4 operations and multicast routing.

Note

The majority of first-generation leaf switches can no longer be ordered from Cisco. All Nexus 9300 Series ACI leaf switches whose model numbers end in PX, TX, PQ, PX-E, and TX-E are considered first-generation leaf switches.

Second-Generation Leaf Switches

Second-generation ACI leaf switches are Nexus 9300 Series platforms that are based on cloud-scale ASICs. Second-generation leaf switches support Remote Leaf and ACI Multi-Tier, have significantly larger policy CAM sizes, and offer enhanced hardware capabilities and port speeds.

Note

MACsec is supported on all ports with speeds greater than or equal to 10 Gbps on Nexus 9300 ACI switches

whose model numbers end in FX. Check specific support levels for other platforms.

ACI leaf switches whose model numbers end in EX, FX, FX2, and FXP are considered second-generation leaf switches.

[Table 2-7](#) provides details about second-generation switches that have 1/10 Gigabit Ethernet copper port connectivity for servers.



Table 2-7 Second-Generation 1/10 Gigabit Ethernet Copper Leaf Switches

Characteristic	Nexus 93108T C-EX	Nexus 9348GC -FXP	Nexus 93108T C-FX	Nexus 93216T C-FX2
Form factor	1 RU fixed	1 RU fixed	1 RU fixed	2 RU fixed
100 Mbps and 1 Gigabit Ethernet copper ports	N/A	48	N/A	N/A
100 Mbps and 1/10 Gigabit Ethernet copper ports	48	N/A	48	96

Characteristic	Nexus 93108T C-EX	Nexus 9348GC -FXP	Nexus 93108T C-FX	Nexus 93216T C-FX2
10/25 Gigabit Ethernet ports	N/A	N/A	4	N/A
40/100 Gigabit Ethernet ports	6	2	6	12
ACI Multi-Pod support	Yes	Yes	Yes	Yes
Remote Leaf support	Yes	Yes	Yes	Yes
Can be used as a Tier 1 leaf	Yes	Yes	Yes	Yes
Can be used as a Tier 2 leaf	Yes	Yes	Yes	Yes

The Nexus 9348GC-FXP switch has 48 ports, offering 100 Mbps or 1 Gigabit Ethernet connectivity. These ports have RJ-45 connections, eliminating the need for transceivers. Due to its low cost and support for cloud-scale features, the

Nexus 9348GC-FXP is an ideal replacement for Fabric Extenders.

Note

Support for ACI Multi-Site is dependent on spine switches in the fabric and not leaf switches. Also, at the time of writing, CloudSec is most relevant to spine switches.

Table 2-8 details second-generation switches that provide 1/10/25 Gigabit Ethernet fiber port connectivity for servers.



Table 2-8 Second-Generation 1/10/25 Gigabit Ethernet Fiber Leaf Switches

Characteristic	Nexus 93180YC-EX	Nexus 93180YC-FX	Nexus 93240YC-FX2	Nexus 93360YC-FX2
Form factor	1 RU fixed	1 RU fixed	1.2 RU fixed	2 RU fixed
1/10/25 Gigabit Ethernet ports	48	48	48	96

40/100 Gigabit Ethernet ports	6	12	12	
ACI Multi-Pod support	Yes	Yes	Yes	Yes
Remote Leaf support	Yes	Yes	Yes	Yes
Can be used as a Tier 1 leaf	Yes	Yes	Yes	Yes
Can be used as a Tier 2 leaf	Yes	Yes	Yes	Yes

[Table 2-9](#) lists details on the only second-generation switch available at the time of writing that provides 40/100 Gigabit Ethernet connectivity for servers.



Table 2-9 Second-Generation 40/100 Gigabit Ethernet Leaf Switches

Characteristic	Nexus 9336C-FX2

Form factor	1 RU fixed
40/100 Gigabit Ethernet ports	36
ACI Multi-Pod support	Yes
Remote Leaf support	Yes
Can be used as a Tier 1 leaf	Yes
Can be used as a Tier 2 leaf	Yes

Exam Preparation Tasks

As mentioned in the section “How to Use This Book” in the Introduction, you have a couple of choices for exam preparation: [Chapter 17, “Final Preparation,”](#) and the exam simulation questions in the Pearson Test Prep Software Online.

Review All Key Topics

Review the most important topics in this chapter, noted with the Key Topic icon in the outer margin of the page. [Table 2-](#)

[10](#) lists these key topics and the page number on which each is found.



Table 2-10 Key Topics for [Chapter 2](#)

Key Topic Element	Description	Page Number
List	Describes APICs, spine switches, and leaf switches	23
List	Describes some functions engineers commonly evaluate when deciding whether to dedicate leaf switches to functions	24
Paragraph	Describes ACI Multi-Pod	25
Paragraph	Calls out requirements for an ACI Multi-Pod IPN	25
	Describes ACI Multi-Site	26

Paragraph		
Paragraph	Explains APIC cluster separation in ACI Multi-Site fabrics and MSO communication with each cluster	27
Paragraph	Calls out requirements for an ACI Multi-Site ISN	27
Paragraph	Explains why ACI Multi-Site requires the use of at least one Gen 2 spine in each site	28
Paragraph	Describes Remote Leaf	30
Paragraph	Explains the significance of sizes in APIC purchases and the relevance of M versus L models	32
Paragraph	Explains APIC hardware generations and correlation with UCS C-Series server generations	32
Table 2-5	Lists first-generation spine switches	37

Table 2-6	Lists second-generation spine switches	38
Table 2-7	Lists second-generation 1/10 Gigabit Ethernet copper leaf switches	39
Table 2-8	Lists second-generation 1/10/25 Gigabit Ethernet fiber leaf switches	40
Table 2-9	Lists second-generation 40/100 Gigabit Ethernet leaf switches	40

Complete Tables and Lists from Memory

There are no memory tables or lists in this chapter.

Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary: [**fabric port**](#)

- [**border leaf**](#)
- [**service leaf**](#)
- [**compute leaf**](#)
- [**IP storage leaf**](#)
- [**stretched ACI fabric**](#)

transit leaf
ACI Multi-Pod
ACI Multi-Site
sharding

Chapter 3

Initializing an ACI Fabric

This chapter covers the following topics:

Understanding ACI Fabric Initialization: This section describes the planning needed prior to fabric initialization and the process of initializing a new ACI fabric.

Initializing an ACI Fabric: This section walks through the process of initializing an ACI fabric.

Basic Post-Initialization Tasks: This section touches on some of the basic tasks often performed right after fabric initialization.

This chapter covers the following exam topics:

- 1.4 Describe ACI fabric discovery
- 5.1 Implement out-of-band and in-band
- 5.3 Implement configuration backup (snapshot/config import export)
- 5.5 Configure an upgrade

Not all ACI engineers will be initializing new fabrics. Some will be more operations focused; others will be more implementation or design focused. But understanding the

fabric discovery and initialization process is important for all ACI engineers.

For operations engineers, there is a possibility that new switch onboarding may necessitate troubleshooting of the switch discovery process. Implementation-focused individuals, on the other hand, may be more interested in understanding the planning necessary to deploy ACI fabrics.

This chapter first reviews the fabric discovery process. It then reviews the steps necessary for initializing an ACI fabric, discovering and onboarding switches, and completing basic post-initialization tasks, such as APIC and switch upgrades.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 3-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Questions.”](#)

Table 3-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding ACI Fabric Initialization	1–4

Initializing an ACI Fabric	5, 6
Basic Post-Initialization Tasks	7-10

Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

- 1.** A company has purchased APICs for an ACI deployment. Which of the following switch platforms is the best candidate for connecting the APICs to the fabric?
 - a.** Nexus 9364C
 - b.** Nexus 9336PQ
 - c.** Nexus 9332C
 - d.** Nexus 93180YC-FX

- 2.** Changing which of the following parameters necessitates a fabric rebuild? (Choose all that apply.)
 - a.** Infrastructure VLAN
 - b.** APIC OOB IP address
 - c.** Fabric ID
 - d.** Active or standby status of a controller

- 3.** At the end of which stage in the switch discovery process are switches considered to be fully activated?
- a.** Switch software upgrades
 - b.** IFM establishment
 - c.** LLDP neighbor discovery
 - d.** TEP IP assignment to nodes
- 4.** An ACI engineer is initializing a fabric, but the first APIC is unable to add a seed switch to the Fabric Membership view. Which of the following could potentially be the causes? (Choose all that apply.)
- a.** No spines have yet been discovered.
 - b.** The active APIC in-band interface connects to an NX-OS switch.
 - c.** The APIC has not received a DHCP Discover message from the seed leaf.
 - d.** The APICs need to form a cluster first.
- 5.** An administrator has made several changes pertinent to the Cisco IMC while bootstrapping an APIC. Which of the following might be preventing fabric discovery?
- a.** The IP address assigned to the Cisco IMC is incorrect.
 - b.** The NIC mode has been updated to Shared LOM.
 - c.** The Cisco IMC default gateway settings is incorrect.
 - d.** The Cisco IMC firmware has been updated.
- 6.** Which of the following is associated exclusively with spine switches?
- a.** VTEP
 - b.** PTEP
 - c.** DTEP

- d.** Proxy-TEP
- 7.** Which of the following import types and modes enables a user to overwrite all current configurations with settings from a backup file?
- a.** Atomic Merge
 - b.** Best Effort Merge
 - c.** Atomic Replace
 - d.** Best Effort Replace
- 8.** Which of the following are valid protocols for forwarding ACI backups to a remote server? (Choose all that apply.)
- a.** TFTP
 - b.** FTP
 - c.** SFTP
 - d.** SCP
- 9.** An administrator wants to conduct an upgrade of an ACI fabric. How can he best group the switches to ensure minimal outage, assuming that servers are dual-homed?
- a.** Create two upgrade groups: one for spines and one for leafs.
 - b.** Create two upgrade groups: one for odd switch node IDs and one for even switch node IDs.
 - c.** Create four upgrade groups and randomly assign node IDs to each.
 - d.** Create four upgrade groups: one for odd leafs, one for even leafs, one for odd spines, one for even spines.
- 10.** True or false: ACI can take automated scheduled backups.
- a.** True

b. False

Foundation Topics

Understanding ACI Fabric Initialization

Before administrators can create subnets within ACI and configure switch ports for server traffic, an ACI fabric needs to be initialized.

The process of fabric initialization involves attaching APICs to leaf switches, attaching leaf switches to spines, configuring APICs to communicate with leaf switches, and activating the switches one by one until the APICs are able to configure all switches in the fabric. Let's look first at the planning needed for fabric initialization.

Planning Fabric Initialization

The planning necessary for fabric initialization can be divided into two categories:

- **Cabling and physical deployment planning:** This category of tasks includes racking and stacking of hardware, cabling, powering on devices, and guaranteeing proper cooling. This book addresses only some of the basic cabling requirements because facilities issues are not the focus of the Implementing Cisco Application Centric Infrastructure DCACI 300-620 exam.
- **Planning of minimal configuration parameters:** This includes preparation of all the configurations

needed to bootstrap the APICs, enable all ACI switches, and join APICs to a cluster.

One way to approach planning an ACI fabric initialization is to create a fabric initialization checklist or a basic table that includes all the information needed to set up the fabric.

Understanding Cabling Requirements

Before initializing a fabric, you need to run cabling between leaf and spine fabric ports. By default, fabric ports are the high-order ports on the right side of leaf switches. They are generally high-bandwidth ports compared to the server downlinks. [Figure 3-1](#) shows a Nexus 93180YC-FX leaf switch. The six ports to the right are all fabric ports by default. The phrase “by default” is intentional here: On leaf switches, fabric ports can be converted to server downlinks and vice versa, but the switch must first be initialized into a fabric.



Figure 3-1 *Nexus 93180YC-FX Leaf with Six Fabric Ports*

Unlike the Nexus 93180YC-FX, a number of leaf platforms have default fabric ports that cannot be easily distinguished by their physical appearance.

Leaf fabric ports can generally be connected to any spine ports (except the spine out-of-band [OOB] management port and any 10 Gbps ports), as long as the transceivers and port speeds are compatible.

Not all leaf-to-spine connections need to be run for fabric discovery to be possible, but there needs to be enough physical connectivity to allow all switches and APICs to have at least a single path to one another.

For example, [Figure 3-2](#) does not represent a full-mesh connectivity between the leaf and spine layers, but it is a perfectly valid topology for the purpose of enabling a full fabric initialization.

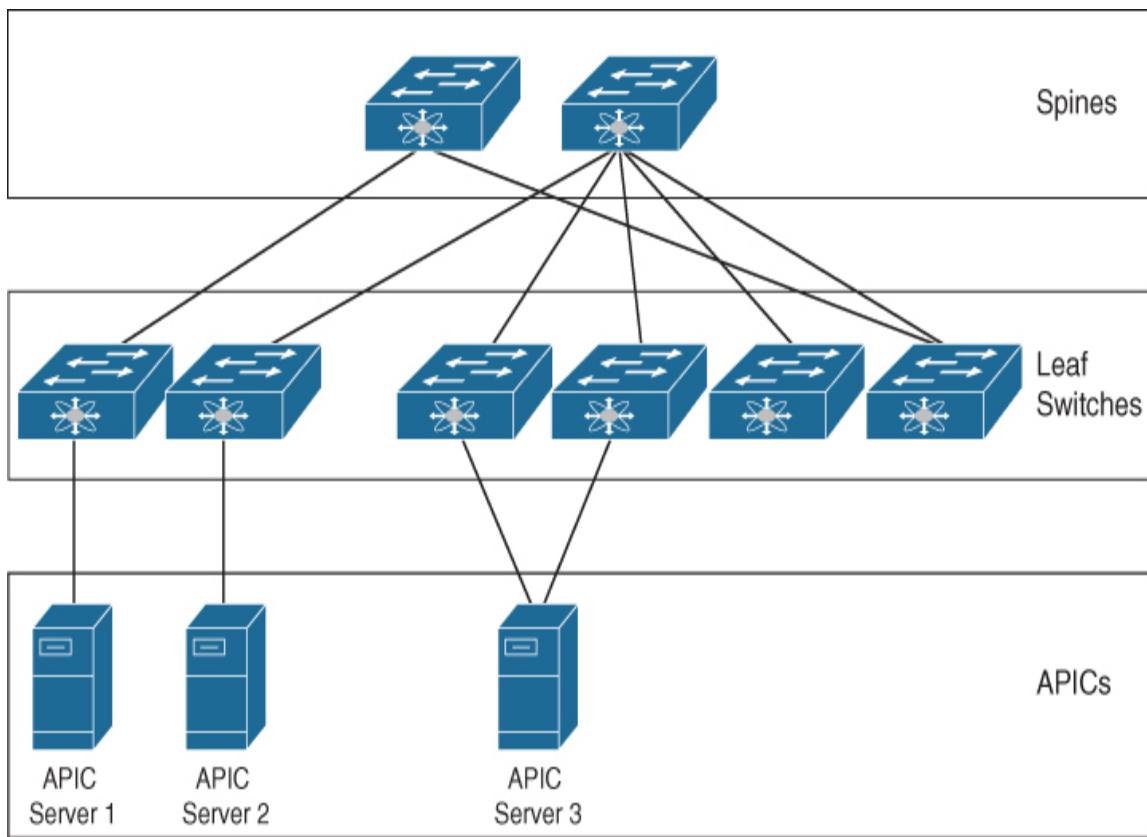


Figure 3-2 Sample Topology Enabling Complete Fabric Discovery

Connecting APICs to the Fabric

In addition to leaf-to-spine fabric port connectivity, the APICs need to be able to establish an in-band communication path through the fabric.

On the back of an APIC, you can see a number of different types of ports. [Figure 3-3](#) shows a rear-panel view of a third-generation APIC populated with a VIC 1455 card.

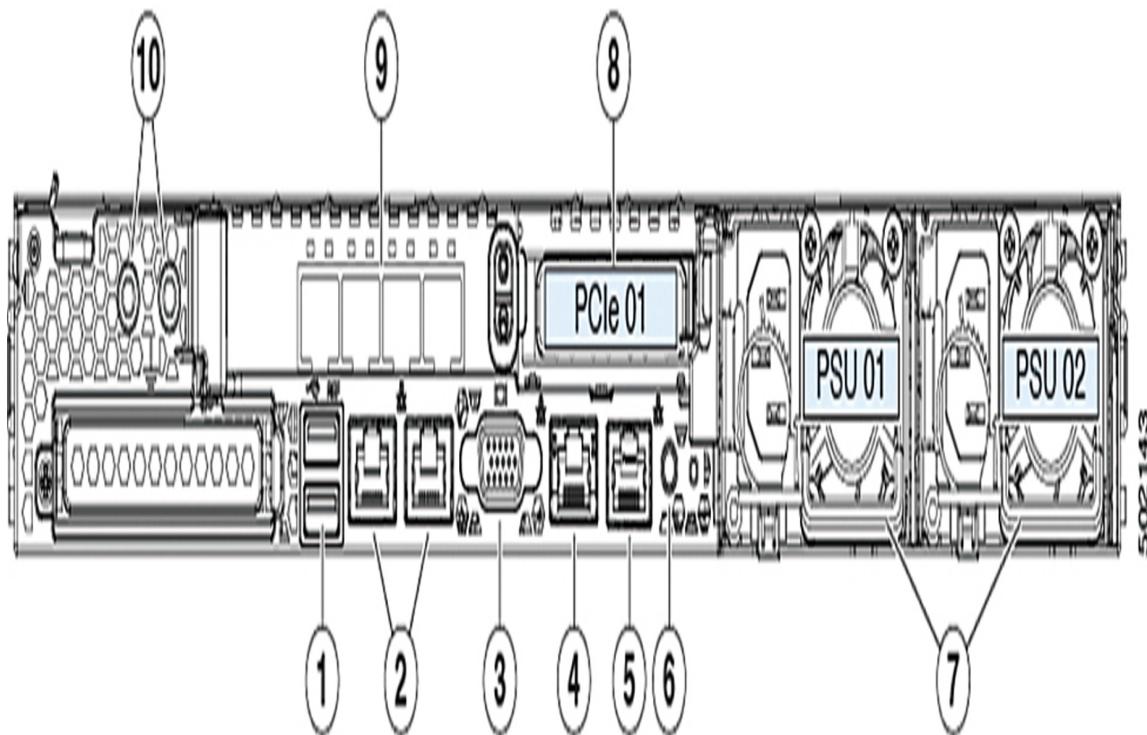


Figure 3-3 Rear View of a Third-Generation APIC

[Table 3-2](#) provides a legend highlighting the components shown in [Figure 3-3](#).

Table 3-2 Legend for Components Numbered in [Figure 3-3](#)

Nu mb er	Component	Nu mb er	Component
1	USB 3.0 ports (2)	6	Rear unit identification button/LED

2	Dual 1 /10 Gigabit Ethernet ports (LAN1 and LAN2)	7	Power supplies (two, redundant as 1+1)
3	VGA video port (DB-15 connector)	8	PCIe riser 1/slot 1 (x16 lane)
4	1 Gigabit Ethernet dedicated management port	9	VIC 1455 with external 10/25 Gigabit Ethernet ports (4)
5	Serial port (RJ-45 connector)	10	Threaded holes for dual-hole grounding lug

Out of the components depicted in [Figure 3-3](#), the VIC 1455 ports are of most importance for the fabric discovery process because they form the in-band communication channel into the fabric. The VIC 1455 card has four 10/25 Gigabit Ethernet ports. VIC adapters in earlier generations of APICs had two 10 Gigabit Ethernet ports instead. At least one VIC port on each APIC needs to be cabled to a leaf to enable full APIC cluster formation. For redundancy purposes, it is best to diversify connectivity from each APIC across a pair of leaf switches by connecting at least two ports.



In first- and second-generation APICs sold with variants of dual-port VIC 1225 cards, ports 1 and 2 would need to be cabled up to leaf switches to diversify connectivity. In third-generation APICs, however, ports 1 and 2 together represent logical port eth2-1, and ports 3 and 4 together represent eth2-2. Ports eth2-1 and eth2-2 are then bundled together into an active/standby team at the operating system level. For this reason, diversifying in-band APIC connectivity across two leaf switches in third-generation APICs requires that one cable be connected to either port 1 or port 2 and another cable be attached to either port 3 or port 4. Connecting both ports that represent a logical port (for example, ports 1 and 2) to leaf switches in third-generation APICs can result in unpredictable failover issues.

Not all ACI leaf switches support 10/25 Gigabit Ethernet cabling. During the deployment planning stage, it is important to ensure that the leaf nodes to which the APICs connect actually support the available VIC port speeds and that proper transceivers and cabling are available.

Initial Configuration of APICs

Out of the box, APICs come with ACI code installed. Normally, switch configuration involves establishing console connectivity to the switch and implementing a basic configuration that allows remote SSH access to the switch. APICs, on the other hand, are servers and not network switches. As such, it is easiest to configure APICs using a crash cart with a standard DB-15 VGA connector and a USB keyboard.

APIC OOB Configuration Requirements

Key Topic

In addition to cabling the in-band communication channel, APICs have two embedded LAN on motherboard (LOM) ports for out-of-band management of the APIC. In third-generation APICs, these dual LAN ports support both 1 and 10 Gigabit Ethernet. (In [Figure 3-3](#), these two LOM ports are shown with the number 2.) As part of the initialization process, users enter an out-of-band IP address for each APIC. The APIC then bonds these two LOM interfaces together and assigns the out-of-band IP address to the bond. From the out-of-band switch to which these ports connect, these connections appear as individual links and should not be misinterpreted as port channels. Basically, the APIC binds the OOB MAC and IP address to a single link and repins the traffic over to the second link if the active interface fails.

Key Topic

OOB management interfaces should not be confused with the Cisco Integrated Management Controller (Cisco IMC) port on the APICs. The [**APIC Cisco IMC**](#) allows lights-out management of the physical server, firmware upgrades, and monitoring of server hardware health. While the dual 1/10 Gigabit Ethernet LOM ports enable out-of-band access to the APIC operating system, the Cisco IMC provides out-of-band access to the server hardware itself. With Cisco IMC access, an engineer can gain virtual KVM access to the server and reinstall the APIC operating system remotely in the event that the APIC is no longer accessible. But the Cisco IMC cannot be used to gain HTTPS access to the ACI management interface. Because of the significance of Cisco IMC in APIC recovery, assigning an IP address to the Cisco

IMC is often viewed as a critically important fabric initialization task.

APIC OOB IP addresses and Cisco IMC IP addresses are often selected from the same subnet even though it is not required for them to be in the same subnet.

Out-of-Band Versus In-Band Management

By default, administrators configure ACI fabrics through the dual OOB interfaces on the APICs. The APICs, in turn, configure switches and communicate with one another using the in-band channel over the VIC adapters.

If the default behavior of managing the fabric through the OOB interfaces is not desirable, administrators can implement in-band management.

There are many factors to consider when determining whether to use in-band management, but the only configuration option available during APIC initialization is to implement OOB management. Administrators can then log in to the ACI GUI and manually implement in-band management.

Out-of-band management of ACI fabrics is the most popular deployment option.

[Chapter 13, “Implementing Management,”](#) discusses in-band management, its implications, and implementation in detail.

Configuration Information for Fabric Initialization

Table 3-3 describes the basic configuration parameters that need to be planned before an ACI fabric can be initialized and that you need to understand for the DCACI 300-620 exam.



Table 3-3 Basic Configuration Parameters for Fabric Initialization

C on fi g ur at io n Pa ra m et er	Description
Fa bri c Na m e	A user-friendly name for the fabric. If no name is entered, ACI uses the name ACI Fabric1.

Fabric ID	A numeric identifier between 1 and 128 for the ACI fabric. If no ID is entered, ACI uses 1 as the fabric ID.
Number of active controllers	A self-explanatory parameter whose valid values are 1 through 9. The default value is 3 for three APICs. If the intent is to add additional APICs to the fabric in the future, select 3 and modify this parameter when it is time to add new APICs.
Pod ID	A parameter that determines the unique pod ID to which the APIC being configured is attached. When ACI Multi-Pod is not being deployed, use the default value 1.

<p>St</p> <p>Co nt rol ler</p>	<p>An APIC added to a fabric solely to aid in fabric recovery and in reestablishing an APIC quorum during a prolonged outage. If the APIC being initialized is a standby APIC, select Yes for this parameter.</p>
<p>Co nt rol ler</p>	<p>The unique ID number for the APIC being configured. Valid values are between 1 and 32. The first three active APICs should always be assigned IDs between 1 and 3. Valid node ID values for standby APICs range from 16 to 32.</p>
<p>Co nt rol ler Na m e</p>	<p>The unique APIC hostname.</p>

Po d 1 TE P Po ol	The TEP pool assigned to the seed pod. A TEP pool is a subnet used for internal fabric communication. This subnet can potentially be advertised outside ACI over an IPN or ISN or when a fabric is extended to virtual environments using the AVS or AVE. TEP pool subnets should ideally be unique across an enterprise environment. Cisco recommends that TEP pool subnet sizes be between /16 and /22. TEP pool sizes <i>do</i> impact pod scalability, and use of /16 or /17 ranges is highly advised. Each pod needs a separate TEP pool. However, during APIC initialization, the TEP pool assigned to the seed pod (Pod 1) is what should be entered in the initialization wizard because all APICs in Multi-Pod environments pull their TEP addresses from the Pod 1 TEP pool.
Infra structure VLAN	The VLAN ID used for control communication between ACI fabric nodes (leaf switches, spine switches, and APICs). The infrastructure VLAN is also used for extending an ACI fabric to AVS or AVE virtual switches. The infra VLAN should be unique and unused elsewhere in the environment. Acceptable IDs are 2 through 4094. Because the VLAN may need to be extended outside ACI, ensure that the selected infrastructure VLAN does not fall into the reserved VLAN range of non-ACI switches.

B The IP address range used for multicast within a fabric.
D In ACI Multi-Site environments, the same range can be
M used across sites. If the administrator does not change
ult the default range, 225.0.0.0/15 will be selected for this
ic parameter. Valid ranges are between 225.0.0.0/15 and
as 231.254.0.0/15. A prefix length of 15 must be used.

t
Ad
dr
es
se
s
(G
iP
o)

AP IC O B Ad dr es se s an d De fa ult Ga te w ay	Addresses assigned to OOB LOM ports for access to the APIC GUI. These ports are separate from the Cisco IMC ports.
Pa ss w or d St re ng th	A parameter that determines whether to enforce the use of passwords of a particular strength for all users. The default behavior is to enforce strong passwords.

Some of the configuration parameters listed in [Table 3-3](#) cannot be changed and require that a fabric be wiped clean

and re-initialized in case of a misconfiguration. Specifically, the parameters to which attention is most critically important include Fabric Name, Fabric ID, Pod 1 TEP Pool, and Infrastructure VLAN.

Switch Discovery Process

Following a minimal configuration bootstrap of the first APIC, switch discovery can begin. So how do APICs use the parameters in [Table 3-3](#) to discover switches and enable them to join the fabric? [Figure 3-4](#) provides a high-level illustration of the process that takes place.

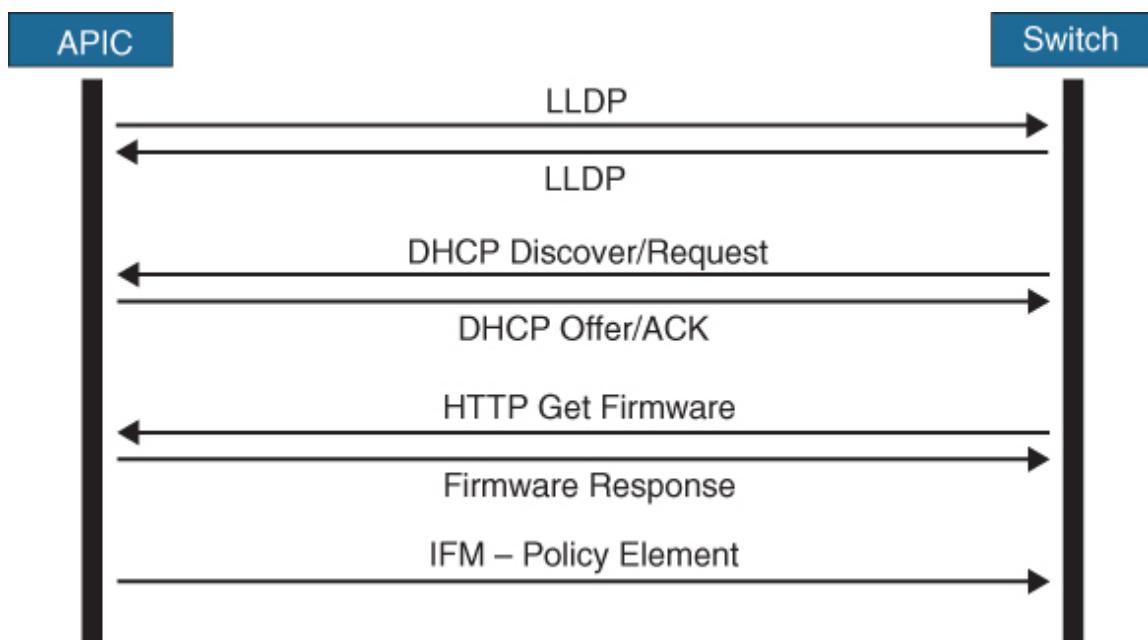


Figure 3-4 *Switch Discovery Process*



The process depicted in [Figure 3-4](#) includes the following steps:

Step 1.LLDP neighbor discovery: After a minimal configuration bootstrap, the first APIC begins sending out LLDP packets on its in-band interfaces. Unregistered leaf switches send LLDP packets on all operational ports. The APIC should eventually pick up LLDP packets from the neighboring leaf if the switch is fully operational and has ACI code installed. From the LLDP packets, the APIC can determine the serial number and hardware platform of the attached device.

Step 2.TEP IP assignment to nodes: In addition to LLDP packets, unregistered ACI switches send DHCP Discover packets on operational interfaces. Once an APIC detects a switch via LLDP and is able to process DHCP Discover packets from the leaf, it adds the device to the Fabric Membership tab. An administrator then needs to register the switch to authorize it to join the fabric. The registration process maps a node ID to the switch and configures its hostname. The switch registration begins with the APIC responding to the switch DHCP requests with a DHCP Offer packet. The leaf confirms that it does want the offered IP address using a DHCP Request message, following which the APIC confirms the IP assignment with a DHCP ACK packet. APICs pull the IP addresses assigned during this process from the TEP pool range configured during APIC initialization. Each leaf switch is assigned a TEP address. These TEP addresses reside in a VRF instance called overlay-1 in a tenant called infra.

Step 3.Switch software upgrades, if necessary: APICs are able to communicate to switches that they need to undergo upgrades to a particular code level before they can be moved into production

status. If a switch upgrade is required, the switch downloads the necessary firmware from the APICs, performs an upgrade, and reboots. The Default Firmware Version setting determines whether a switch upgrade is necessary. This setting is detailed later in this chapter.

Step 4. Policy element intra-fabric messaging (IFM)

setup: After the switch boots up with the intended code revision, the APIC authenticates the switch by using the switch certificate signed at the factory and opens communication with the switch TEP address over the infrastructure VLAN using ***intra-fabric messaging (IFM)***. All IFM channel communication over the infrastructure VLAN is encrypted using TLS Version 1.2, and every message that comes to the switch over the IFM channel must be decrypted before it is processed by the switch. Once APICs establish IFM communication with a switch, the switch is fully activated. Any policy push from the APICs to switches rides this encrypted IFM communication channel.

Depending on the switch being discovered, some minor tasks may be added to the overall discovery process. For example, a Remote Leaf discovery would additionally require DHCP relay functionality to be enabled for DHCP packets from the Remote Leaf to reach the APICs. (The task of enabling DHCP relay does not conflict with the four primary steps outlined for switch discovery.) Another example of minor tasks added to the process is establishment of IS-IS adjacencies between leaf and spine switches using the switch loopback 0 interfaces.

Fabric Discovery Stages

After the bootstrapping of the first APIC, fabric initialization happens in the following three phases:



- 1. Seed leaf initialization:** Even when an APIC VIC adapter attaches to two or more operational leaf switches, the APIC can detect only one of the leaf switches. This is because APIC VIC adapters operate in active/standby mode. Activation of the first leaf switch by an administrator allows the leaf to function as a seed switch for further discovery of the fabric.
- 2. Spine initialization:** After the seed leaf initialization, any spines with fabric ports attached to the seed leaf are detected and added to the Fabric Membership view to allow spine activation.
- 3. Initialization of leaf switches and additional APICs:** As spines are brought into the fabric, ACI can detect other leaf switches connected to them. Administrators can then activate the leaf switches. Once the leaf switches connected to additional APICs join the fabric, the APIC cluster forms, and APIC synchronization begins. Controllers join the cluster based on node ID. In other words, the third APIC (whose node ID is 3) joins the cluster only after the first and second APICs have joined. If any critical bootstrap configuration parameters have been entered incorrectly on the additional controllers, the APIC fails to join the cluster and needs to be wiped clean and re-initialized.

Note that the phases outlined here describe cluster formation as part of the final leaf initialization phase. However, if active in-band interfaces on all APICs connect to

the seed leaf switch, the APIC cluster can form during the seed leaf initialization phase.

Switch Discovery States

During the discovery process, switches transition between various states. [Table 3-4](#) describes the different discovery states.



Table 3-4 Fabric Node Discovery States

State Description	
Unknown	The node has been detected, but a node ID has not yet been assigned by an administrator in the Fabric Membership view.
Undiscovered	An administrator has prestaged a switch activation by manually mapping a switch serial number to a node ID, but a switch with the specified serial number has not yet been detected via LLDP and DHCP
Discovering	The node has been detected, and the APICs are in the process of mapping the specified node ID as well as a TEP IP address to the switch.

Unsup	The node is a Cisco switch, but it is not supported or ported. The firmware version is not compatible with the ACI fabric.
Disabled/Decommissioned	The node has been discovered and activated, but a user disabled or decommissioned it. The node can be reenabled.
Maintenance	An ACI administrator has put the switch into maintenance mode (graceful insertion and removal).
Inactive	The node has been discovered and activated, but it is not currently accessible. For example, it may be powered off, or its cables may be disconnected.
Active	The node is an active member of the fabric.

Initializing an ACI Fabric

Once all cabling has been completed and the APICs and ACI switches have been turned on, it is time to initialize the fabric. The tasks in this section lead to the configuration of the APIC Cisco IMC addresses, the initialization of the APICs, and the activation of ACI switches.

Changing the APIC BIOS Password

One of the things ACI implementation engineers usually do during APIC setup is to change the default BIOS password.

To change the BIOS password, you press the F2 key during the boot process to enter the BIOS setup. Then you can enter the default BIOS password **password** in the Enter Password dialog box and navigate to the Security tab, choose Set Administrator Password, and enter the current password in the Enter Current Password dialog box. When the Create New Password dialog box appears, enter the new password and then enter the new password again in the Confirm New Password dialog box. Finally, navigate to the Save & Exit tab and choose Yes in the Save & Exit Setup dialog box. The next time BIOS setup is accessed, the new BIOS password will be needed.

Configuring the APIC Cisco IMC

After changing the BIOS password, it is a good idea to configure a static IP address for the APIC Cisco IMC addresses.

To configure a static IP address for remote Cisco IMC access, press the F8 key during the boot process to enter Cisco IMC. Enter the desired IP addressing details in the section IP (Basic), as shown in [Figure 3-5](#). Then press the F10 key to save the Cisco IMC configuration and wait up to 20 seconds for the configuration change to take effect before rebooting the server.

```
Cisco IMC Configuration Utility Version 2.0 Cisco Systems, Inc.  
*****  
NIC Properties  
NIC mode NIC redundancy  
Dedicated: [X] None: [X]  
Shared LOM: [ ] Active-standby: [ ]  
Cisco Card: Active-active: [ ]  
Riser1: [ ] VLAN (Advanced)  
Riser2: [ ] VLAN enabled: [ ]  
MLom: [ ] VLAN ID: 1  
Shared LOM Ext: [ ] Priority: 0  
IP (Basic)  
IPV4: [X] IPV6: [ ]  
DHCP enabled: [ ]  
CIMC IP: 172.23.142.101  
Prefix/Subnet: 255.255.248.0  
Gateway: 172.23.136.1  
Pref DNS Server:  
  
*****  
<Up/Down>Selection <F10>Save <Space>Enable/Disable <F5>Refresh <ESC>Exit  
<F1>Additional settings
```

Figure 3-5 Enter IP Addressing Details for Cisco IMC

Key Topic

As a best practice, do not modify the NIC Mode or NIC Redundancy settings in Cisco IMC. If there are any discovery issues, ensure that Cisco IMC has been configured with the default NIC Mode setting Dedicated and not Shared. The NIC Redundancy setting should also be left at its default value None.

Initializing the First APIC

When the APIC boots up, basic configuration parameters need to be entered in line with the pre-installation data captured in earlier steps. [Example 3-1](#) shows how the first APIC in a fabric with ID 1 and the name DC1-Fabric1 might be configured. Note that you can leave certain parameters

at their default values by pressing the Enter key without modifying associated values. The BD multicast addresses range, for instance, is left at its default value of 225.0.0.0/15 in the following example.

Example 3-1 Initialization of First APIC

[Click here to view code image](#)

```
Cluster configuration ..  
Enter the fabric name [ACI Fabric1]: DC1-Fabric1  
Enter the fabric ID (1-128) [1]: 1  
Enter the number of active controllers in the fabric (1-9)  
[3]: 3  
Enter the POD ID (1-9) [1]: 1  
Is this a standby controller? [NO]: NO  
Enter the controller ID (1-3) [1]: 1  
Enter the controller name [apic1]: DC1-APIC1  
Enter address pool for TEP addresses [10.0.0.0/16]:  
10.233.44.0/22  
Note: The infra VLAN ID should not be used elsewhere in  
your environment  
and should not overlap with any other reserved VLANs  
on other platforms.  
Enter the VLAN ID for infra network (2-4094): 3600  
Enter address pool for BD multicast addresses (GIP0)  
[225.0.0.0/15]:  
  
Out-of-band management configuration ..  
Enable IPv6 for Out of Band Mgmt Interface? [N]:  
Enter the IPv4 address [192.168.10.1/24]: 172.23.142.29/21  
Enter the IPv4 address of the default gateway [None]:  
172.23.136.1  
Enter the interface speed/duplex mode [auto]:
```

```
admin user configuration ..  
  Enable strong passwords? [Y]:  
  Enter the password for admin:  
  
  Reenter the password for admin:  
  
Cluster configuration ..  
  Fabric name: DC1-Fabric1  
  Fabric ID: 1  
  Number of controllers: 3  
  Controller name: DC1-APIC1  
  POD ID: 1  
  Controller ID: 1  
  TEP address pool: 10.233.44.0/22  
  Infra VLAN ID: 3600  
  Multicast address pool: 225.0.0.0/15
```

```
Out-of-band management configuration ..  
  Management IP address: 172.23.142.29/21  
  Default gateway: 172.23.136.1  
  Interface speed/duplex mode: auto
```

```
admin user configuration ..  
  Strong Passwords: Y  
  User name: admin  
  Password: *****
```

The above configuration will be applied ..

Warning: TEP address pool, Infra VLAN ID and Multicast address pool
cannot be changed later, these are permanent until the fabric is wiped.

Would you like to edit the configuration? (y/n) [n]:

After you complete the minimal configuration bootstrap for the first controller, the APIC starts various services, and the APIC web GUI eventually becomes accessible via the APIC out-of-band management IP address. [Figure 3-6](#) shows the ACI login page. By default, APICs allow web access via HTTPS and not HTTP.

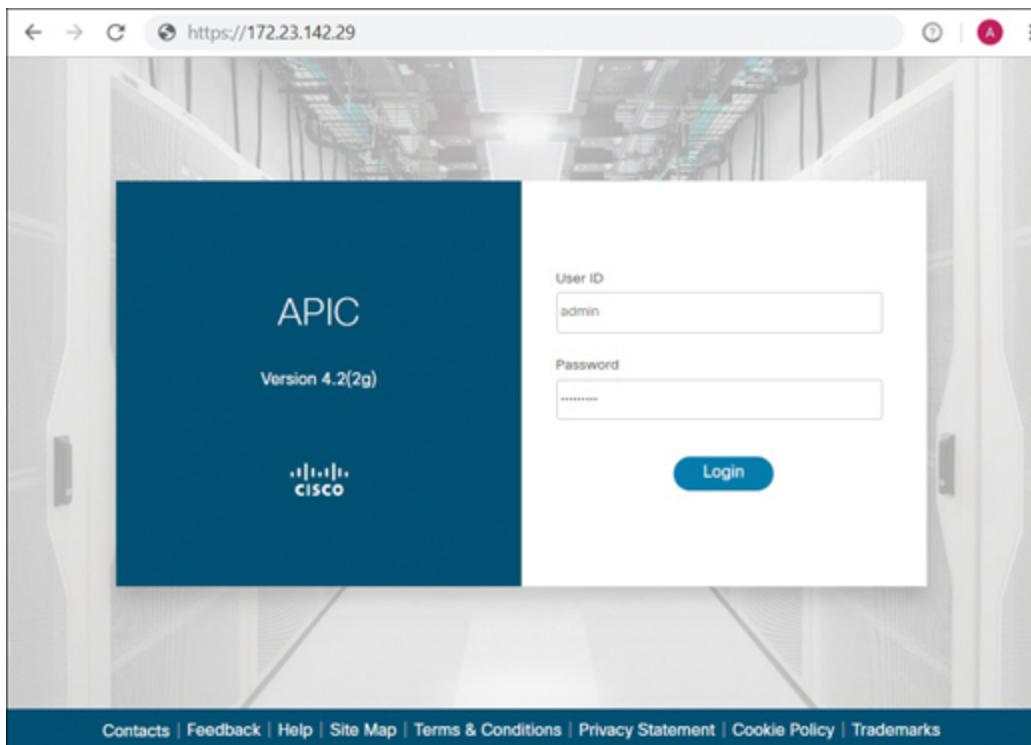


Figure 3-6 The Default ACI Login Screen

Enter **admin** as the username along with the password entered during setup to log in to the APIC.

Discovering and Activating Switches

The switch activation process involves selection of node IDs for all switches. The first three active APICs need to be

assigned node IDs 1, 2, and 3. ACI design engineers have more flexibility in the selection of switch node IDs. As of ACI Release 4.2, valid switch node IDs are between 101 and 4000. Node IDs are cornerstones of ACI stateless networking. Once a switch is commissioned, node ID changes require that the node be decommissioned and cleanly rebooted.

[Figure 3-7](#) shows a hypothetical node ID selection scheme in which spine switches have node ID numbers between 201 and 299 and leaf switches have node numbers between 101 and 199. It is a Cisco best practice to assign subsequent node IDs to leaf switches that are paired into a VPC domain.

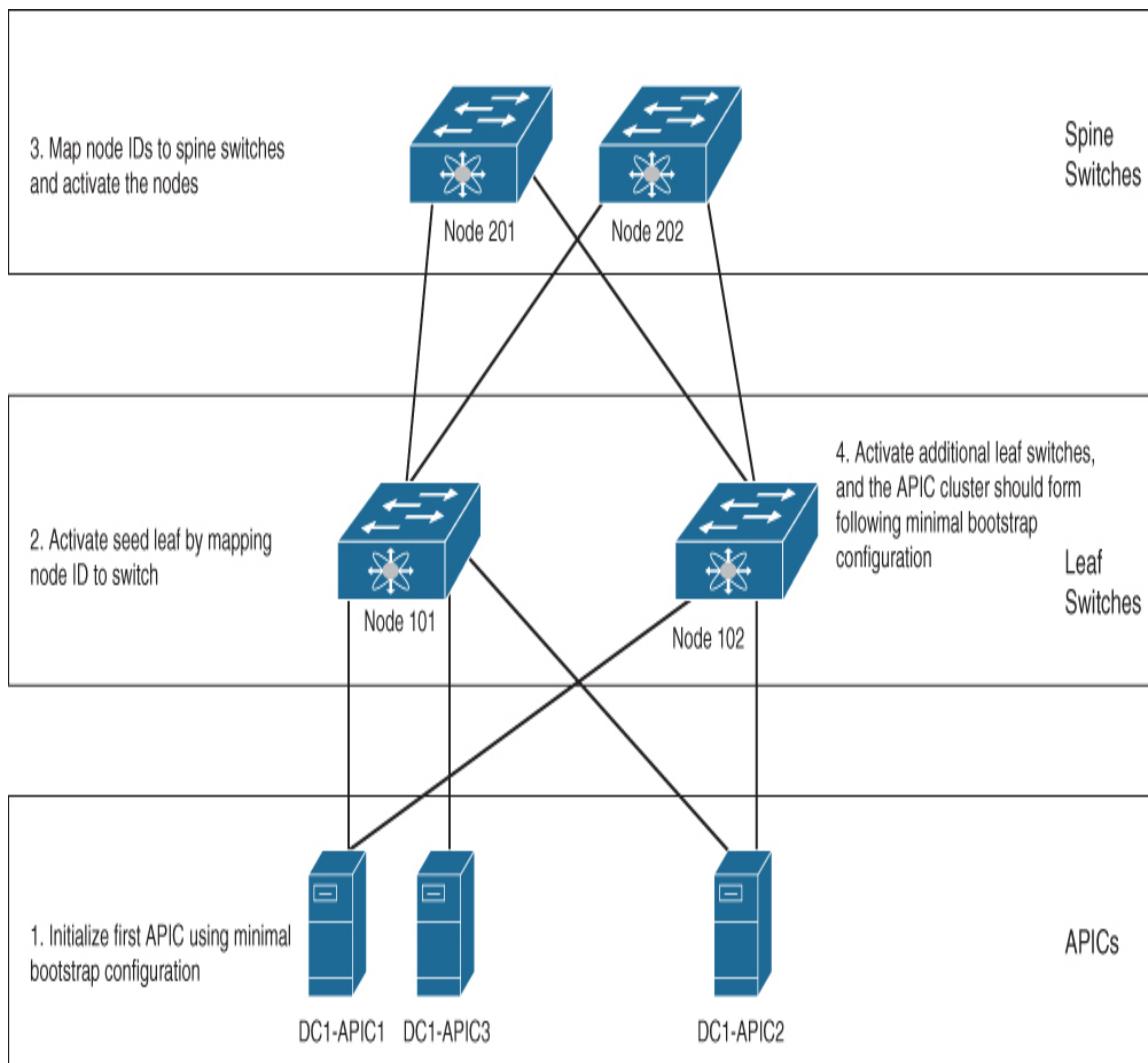


Figure 3-7 Node ID Assignment in a Topology Under Discovery

Following the initialization of DC1-APIC1 in [Figure 3-7](#), the APIC should detect that a leaf switch is connected to its active VIC interface and add it to the Fabric Membership view. Navigate to Fabric, select Inventory, and then click on Fabric Membership. In the Fabric Membership view, select Nodes Pending Registration, right-click the detected switch entry, and select Register, as demonstrated in [Figure 3-8](#). This first leaf switch added to the fabric will serve as the seed leaf for the discovery of the remaining switches in the fabric.

The screenshot shows the Cisco APIC web interface. The top navigation bar includes the Cisco logo, APIC, admin, and various icons. The main menu has tabs for System, Tenants, Fabric (selected), Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. The left sidebar under Inventory lists Quick Start, Topology, Pod 1, Pod Fabric Setup Policy, Fabric Membership (selected), Disabled Interfaces and Decorators, and Duplicate IP Usage. The central content area is titled 'Fabric Membership' and shows three categories: Registered Nodes (0), Nodes Pending Registration (selected, showing 1 node), Unreachable Nodes (0), and Unman (0). Below this, a table lists nodes with columns: Serial Number, Pod ID, Node ID, RL TEP Pool, Name, Node Type, Suppo Model, SSL Certific, and Status. One row for 'FDO2...' is selected, showing values: 1, 0, 0, leaf, yes, n/a. A 'Register' button is highlighted in blue at the bottom of this row. Other buttons include 'Edit Node and Rack Names' and 'Remove From Controller'.

Figure 3-8 Selecting the Entry for Unknown Switch and Launch Registration Wizard

In the node registration wizard, enter values in the fields Pod ID, Node ID, and Node Name (hostname) and then click Register (see [Figure 3-9](#)). If the switch has been auto-

detected by ACI, the role should be auto-populated. The Rack Name parameter is optional. The RL TEP Pool field should be populated only during configuration of a Remote Leaf switch.

The screenshot shows a registration form titled "Register". It includes fields for Serial Number (FDO2), Pod ID (1), Node ID (101), RL TEP Pool (0), Role (leaf), Node Name (LEAF101, highlighted in blue), and Rack Name (select). There are "Cancel" and "Register" buttons at the bottom.

Serial Number:	FDO2
Pod ID:	1
Node ID:	101
RL TEP Pool:	0
Role:	leaf
Node Name:	LEAF101
Rack Name:	select

Cancel Register

Figure 3-9 The Node Registration Wizard

Aside from the leaf and spine roles, the node registration wizard allows assignment of *virtualleaf* and *virtualspine* roles for vPOD switches, the *controller* role for APICs, the *remoteleaf* role, and *tier-2-leaf* role for Tier 2 leaf switches.

Minutes after registering the seed switch, it should move into an active state. The state of commissioned fabric nodes can be verified under the Status column in the Registered Nodes subtab of the Fabric Membership menu.

Figure 3-10 shows that all node IDs depicted in Figure 3-7 earlier in this chapter have been initialized one by one and

have moved to an active state, completing the fabric initialization process.

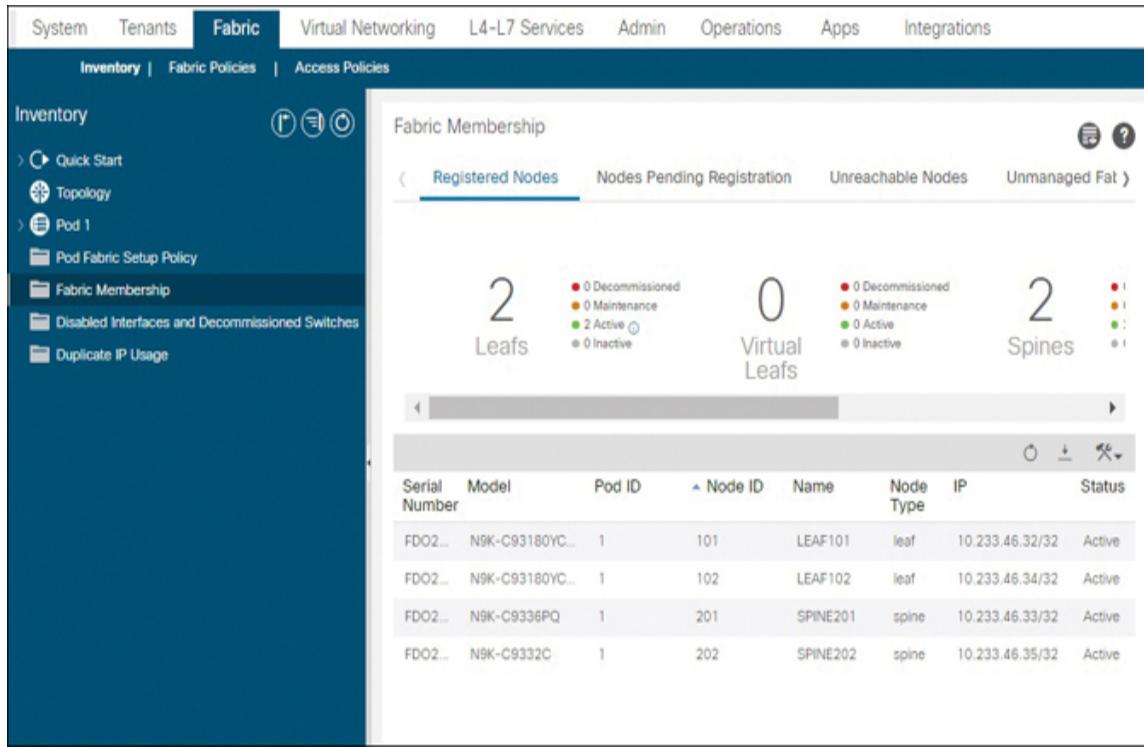


Figure 3-10 Registered Nodes Submenu of the Fabric Membership View

Understanding Graceful Insertion and Removal (GIR)

Figure 3-11 shows that one of the menu options that appears when you right-click a fabric node is Maintenance (GIR). Moving a switch into maintenance mode simulates an uplink failure from the perspective of downstream servers. This feature enables a more graceful way of moving a switch out of the data plane forwarding topology when minor maintenance or switch upgrades are necessary.

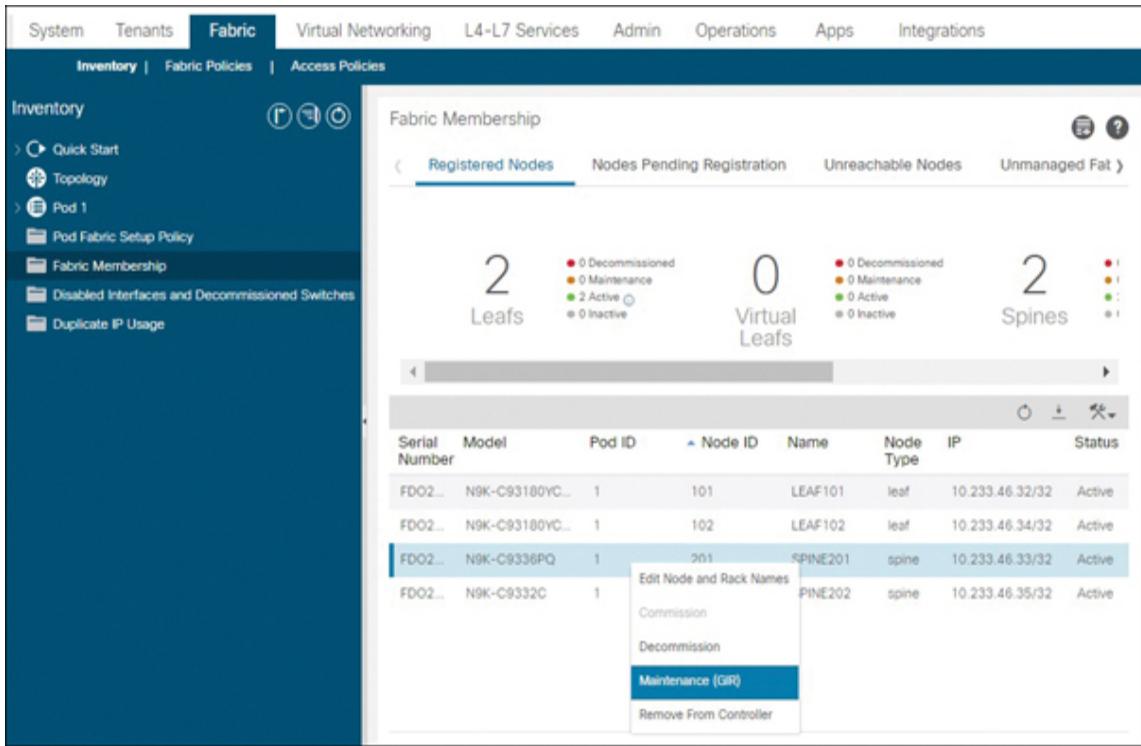


Figure 3-11 Graceful Insertion and Removal Feature

Initializing Subsequent APICs

The minimal configuration bootstrap for subsequent APICs can be performed simultaneously with the initialization of the first APIC. However, the APICs do not form a complete cluster until the end-to-end path between the APICs has been established over the infrastructure VLAN.

Remember that even when multiple APICs have connections to the seed leaf switch, it is still possible that they may not be able to form a cluster through the one seed leaf due to the active/standby status of the VIC adapter interfaces at the time of initialization.

But beyond the process and order of node activation, there is also the issue of bootstrapping requirements to form a cluster. If the fabric ID, fabric name, or Pod 1 TEP pool configured on the subsequent APICs are not the same as

what has been configured for the initial controller, the APIC cluster will never form. In such cases, when the underlying problem is a misconfiguration on the second or third APIC, that APIC needs to be wiped clean and re-initialized. If the first APIC has been misconfigured, the entire fabric needs to be wiped clean and re-initialized.

Some APIC configuration parameters that should not be the same as those entered for the initial APIC include the out-of-band IP address and the APIC node ID.

After establishing end-to-end connectivity, you can verify the health of an APIC cluster by navigating to the System menu, selecting Controllers, opening the Controllers folder, double-clicking an APIC, and then selecting Cluster as Seen by Node. If the controllers are healthy and fully synchronized, all APICs should display Fully Fit in the Health State column, as shown in [Figure 3-12](#).

The screenshot shows the Cisco ACI Controller Manager interface. The top navigation bar includes System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below this is a secondary navigation bar with QuickStart, Dashboard, Controllers, System Settings, Smart Licensing, Faults, Config Zones, Events, Audit Log, and Active Sessions. The main content area is titled "Cluster as Seen by Node". On the left, there's a sidebar under "Controllers" with options like Quick Start, Topology, and a detailed view for "apic1 (Node-1)" which includes Cluster as Seen by Node, Interfaces, Storage, NTP Details, Equipment Fans, Power Supply Units, Equipment Sensors, Memory Slots, Processes, Containers, and APIC-X. The main panel displays "Properties" for the cluster, including Fabric Name: DC1-Fabric1, Target Size: 3, Current Size: 3, and a dropdown for ACI Fabric Intermode Secure Authentication Communications set to "Permissive". Below this is a table titled "Active Controllers" with columns: ID, Name, IP, Admin State, Operational State, Health State, Failover Status, Serial Num, and SSL Certificate. The table contains three rows:

ID	Name	IP	Admin State	Operational State	Health State	Failover Status	Serial Num	SSL Certificate
1	apic1	10.233.44.1	In Service	Available	Fully Fit	idle	FC...	yes
2	apic2	10.233.44.2	In Service	Available	Fully Fit	idle	FC...	yes
3	vapiC3	10.233.44.3	In Service	Available	Fully Fit	idle	F51...	yes

At the bottom right are "Reset" and "Submit" buttons.

Figure 3-12 Verifying Health and Synchronization Status of APICs

Understanding Connectivity Following Switch Initialization

What actually happens during the switch node activation process from a routing perspective? One of the first things that happens is that IS-IS adjacencies are established between the leaf and spine switches, as shown in [Example 3-2](#). Here, interfaces Ethernet 1/49 and 1/50 are the leaf fabric ports.

Example 3-2 Verifying IS-IS Adjacencies Within the Fabric
[Click here to view code image](#)

```
LEAF101# show isis adjacency detail vrf overlay-1
IS-IS process: isis_infra VRF:overlay-1
IS-IS adjacency database:
System ID          SNPA          Level  State   Hold Time
Interface
212E.E90A.0000    N/A           1       UP      00:01:01
Ethernet1/49.34
Up/Down transitions: 1, Last transition: 21d17h ago
Circuit Type: L1
IPv4 Address: 10.233.46.33
232E.E90A.0000    N/A           1       UP      00:00:55
Ethernet1/50.35
Up/Down transitions: 1, Last transition: 21d17h ago
Circuit Type: L1
IPv4 Address: 10.233.46.35
```

A look at the addresses with which LEAF101 has established adjacencies indicates that IS-IS adjacencies are sourced and

destined from and to loopback 0 interfaces on leaf and spine switches. Furthermore, loopback 0 interfaces get associated with all operational fabric ports, as indicated in [Example 3-3](#). The IP address ACI assigns to the loopback 0 interface of a given switch is a specific type of TEP address referred to as a ***physical tunnel endpoint (PTEP)*** address.

Example 3-3 Verifying Switch TEP Addresses

[Click here to view code image](#)

```
LEAF101# show ip int brief | grep -E "lo0|unnumbered"
eth1/49.34          unnumbered           protocol-up/link-
                     up/admin-up
                           (lo0)
eth1/50.35          unnumbered           protocol-up/link-
                     up/admin-up
                           (lo0)
lo0                 10.233.46.32/32    protocol-up/link-
                     up/admin-up

SPINE201# show ip int brief | grep -E "lo0|unnumbered"
eth1/1.37          unnumbered           protocol-up/link-
                     up/admin-up
                           (lo0)
eth1/2.38          unnumbered           protocol-up/link-
                     up/admin-up
                           (lo0)
lo0                 10.233.46.33/32    protocol-up/link-
                     up/admin-up

SPINE202# show ip int brief | grep -E "lo0|unnumbered"
eth1/1.35          unnumbered           protocol-up/link-
                     up/admin-up
                           (lo0)
eth1/2.36          unnumbered           protocol-up/link-
                     up/admin-up
```

	(lo0)	
lo0	10.233.46.35/32	protocol-up/link-up/admin-up

In addition to loopback 0 interfaces, ACI creates loopback 1023 interfaces on all leaf switches. A loopback 1023 interface is used for assignment of a single fabricwide pervasive IP address called a **fabric tunnel endpoint (FTEP)** address. The FTEP address represents the entire fabric and is used to encapsulate traffic in VXLAN to an AVS or AVE virtual switch, if present.

ACI also assigns an SVI and IP address to leaf switches in the infrastructure VLAN. In [Example 3-4](#), internal VLAN 8 on LEAF101 actually maps to VLAN 3600, which is the infrastructure VLAN configured during fabric initialization. Note that the infrastructure VLAN SVI should contain the same IP address for all leaf switches.

Example 3-4 Additional Auto-Established Connectivity in the Overlay-1 VRF Instance

[Click here to view code image](#)

```
LEAF101# show ip int brief vrf overlay-1
(...output truncated for brevity...)
IP Interface Status for VRF "overlay-1"(4)
Interface          Address          Interface Status
eth1/49            unassigned      protocol-up/link-
                           up/admin-up
eth1/49.34         unnumbered     protocol-up/link-
                           up/admin-up
                           (lo0)
eth1/50            unassigned      protocol-up/link-
                           up/admin-up
eth1/50.35         unnumbered     protocol-up/link-
```

```

up/admin-up
          (lo0)
vlan8      10.233.44.30/27   protocol-up/link-
up/admin-up
lo0        10.233.46.32/32   protocol-up/link-
up/admin-up
lo1023    10.233.44.32/32   protocol-up/link-
up/admin-up

```

LEAF101# **show vlan extended**

VLAN	Name	Encap
Ports		
-----	-----	-----
-----	-----	-----
8	infra:default	vxlan-16777209,
Eth1/1, Eth1/2, Eth1/47		vlan-3600

Once an ACI fabric has been fully initialized, each switch should have ***dynamic tunnel endpoint (DTEP)*** entries that include PTEP addresses for all other devices in the fabric as well as entries pointing to spine proxy (*proxy TEP*) addresses. [Example 3-5](#) shows DTEP entries from the perspective of LEAF101 with the proxy TEP addresses highlighted.

Example 3-5 Dynamic Tunnel Endpoint (DTEP) Database

[Click here to view code image](#)

```

LEAF101# show isis dteps vrf overlay-1

IS-IS Dynamic Tunnel End Point (DTEP) database:
DTEP-Address      Role      Encapsulation     Type

```

10.233.46.33	SPINE	N/A	PHYSICAL
10.233.47.65	SPINE	N/A	PHYSICAL, PROXY-
ACAST-MAC			
10.233.47.66	SPINE	N/A	PHYSICAL, PROXY-
ACAST-V4			
10.233.47.64	SPINE	N/A	PHYSICAL, PROXY-
ACAST-V6			
10.233.46.34	LEAF	N/A	PHYSICAL
10.233.46.35	SPINE	N/A	PHYSICAL

If a leaf switch knows the destination leaf behind which an endpoint resides, it is able to tunnel the traffic directly to the destination leaf without using resources on the intermediary spine switches. If a leaf switch does not know where the destination endpoint resides, it can forward the traffic to the spine proxy addresses, and the recipient spine can then perform a lookup in its local Council of Oracle Protocol (COOP) database and forward the traffic to the intended recipient leaf. This spine proxy forwarding behavior is more efficient than forwarding via broadcasts and learning destination switches through ARP. Reliance on spine proxy forwarding instead of flooding of broadcast, unknown unicast, and multicast traffic is called *hardware proxy* forwarding. The benefit of using hardware proxy forwarding is that ACI is able to potentially eliminate flooding within the fabric, allowing the fabric to better scale while also limiting the amount of traffic servers need to process.

Because ACI leaf switches are able to use IS-IS to dynamically learn all PTEP and spine proxy addresses within the fabric, they are able to create tunnel interfaces to various destinations in the fabric. A tunnel in ACI can be simply interpreted as a reference to the next-hop addresses to reach a particular destination. [Example 3-6](#) lists the tunnels on LEAF101. Tunnels 1, 3, and 4 are destined to leaf

and spine PTEP addresses. Tunnels 5 through 7 reference proxy TEP addresses. Finally, tunnels 8, 9, and 10 refer to the TEP addresses assigned to APIC 1, APIC 2, and APIC 3, respectively.

Example 3-6 *Tunnel Interfaces Sourced from lo0 with Different Destinations*

[Click here to view code image](#)

```
LEAF101# show interface tunnel 1-20 | grep -E
' destination|up'
Tunnel1 is up
    Tunnel destination 10.233.46.33
Tunnel3 is up
    Tunnel destination 10.233.46.34
Tunnel4 is up
    Tunnel destination 10.233.46.35
Tunnel5 is up
    Tunnel destination 10.233.47.65
Tunnel6 is up
    Tunnel destination 10.233.47.66
Tunnel7 is up
    Tunnel destination 10.233.47.64
Tunnel8 is up
    Tunnel destination 10.233.44.1
Tunnel9 is up
    Tunnel destination 10.233.44.2
Tunnel10 is up
    Tunnel destination 10.233.44.3
```

Note that aside from IS-IS, ACI enables COOP functionality on all available spine switches as part of the fabric initialization process. This ensures that leaf switches can communicate endpoint mapping information (location and

identity) to spine switches. However, fabric initialization does not result in the automatic establishment of control plane adjacencies for protocols such as MP-BGP. As of the time of this writing, a BGP autonomous system number needs to be selected, and at least one spine has to be designated as a route reflector before MP-BGP can be effectively used within an ACI fabric.

Basic Post-Initialization Tasks

After the initialization of APICs and switches, there are a number of tasks that are generally seen as basic prerequisites for putting a fabric into production. This section gives a rundown of such tasks.

Assigning Static Out-of-Band Addresses to Switches and APICs

Assigning out-of-band addresses to switches ensures that administrators can access switches via SSH. Out-of-band addresses can be assigned statically by an administrator or dynamically out of a pool of addresses.



Figure 3-13 shows how to assign static out-of-band addresses to fabric nodes through the Create Static Node Management Addresses page. To create static out-of-band addresses for a node, navigate to Tenants and select the tenant named mgmt. Within the tenant, double-click the Node Management Addresses folder. Then right-click the Static Node Management Addresses folder and select Create Static Node Management Addresses. Select default from the Out-of-Band Management EPG drop-down box. Chapter 5,

[“Tenants Building Blocks,”](#) describes EPGs thoroughly, but for now you just need to know that the default out-of-band management EPG is an object that represents one or more out-of-band subnets or specific out-of-band addresses used for ACI switches and APICs. Out-of-band management EPGs other than the default object can be created if desired to enable application of granular security policies to different nodes. After entering the node ID in both Node Range fields, the out-of-band IPv4 address, and the out-of-band IPv4 gateway details for a given switch or APIC, click Submit. The static node address mapping should then appear under the Static Node Management Addresses folder.

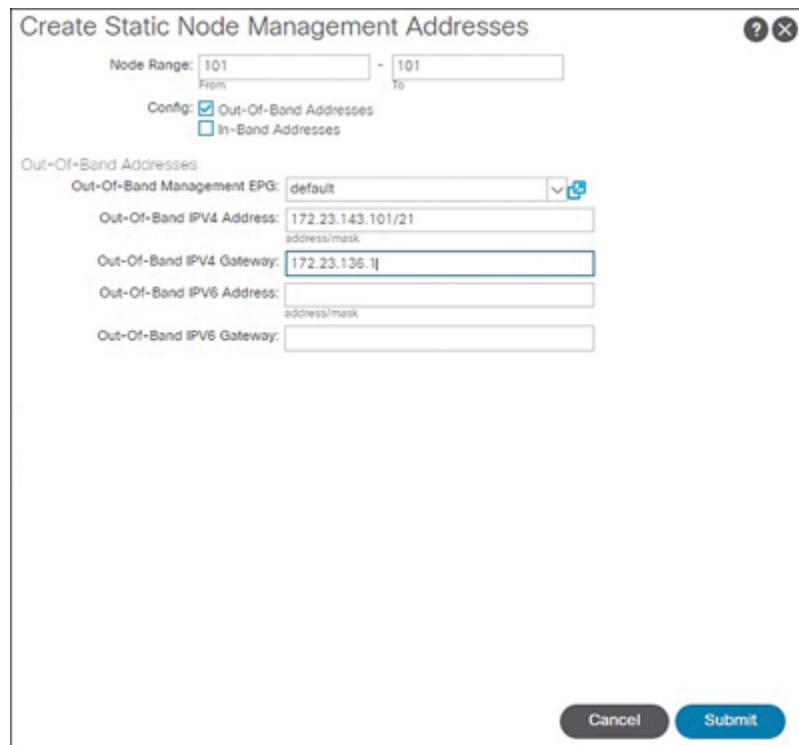


Figure 3-13 Creating Static Out-of-Band Addresses for Switches and APICs

Even though APIC addresses are manually assigned through the controller initialization process, they do not by default appear under the Static Node Management Addresses folder. This is a problem if monitoring solutions such as

SNMP are used to query ACI. Assigning static out-of-band addresses to APICs following fabric initialization helps ensure that certain monitoring functions work as expected. [Figure 3-14](#) illustrates how the original node IDs used during APIC initialization should be used to add static OOB IP addressing entries to APICs.

A screenshot of the Cisco Application Centric Infrastructure (ACI) interface. The top navigation bar includes tabs for System, Tenants (selected), Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below the tabs, there are links for ALL TENANTS, Add Tenant, and Tenant Search. A sidebar on the left shows a tree structure under the "mgmt" tenant, including sections like Quick Start, mgmt, Application Profiles, Networking, IP Address Pools, Contracts, Policies, Services, Node Management EPGs, External Management Network Instance Profiles, Node Management Addresses (selected), Static Node Management Addresses (selected), default, and Managed Node Connectivity Groups. The main content area is titled "Static Node Management Addresses". It features a table with columns: Node ID, Name, Type, EPG, IPV4 Addr, IPV4 Gate, IPV6 Address, and IPV6 Gateway. The table lists several entries: pod-1/node-1 (apic1, Out-Of-Band, default, 1...), pod-1/node-2 (apic2, Out-Of-Band, default, 1...), pod-1/node-3 (vapic3, Out-Of-Band, default, 1...), pod-1/node-101 (LEAF101, Out-Of-Band, default, 1...), pod-1/node-102 (LEAF102, Out-Of-Band, default, 1...), pod-1/node-201 (SPINE201, Out-Of-Band, default, 1...), and pod-1/node-202 (SPINE202, Out-Of-Band, default, 1...). The row for pod-1/node-3 (vapic3) is highlighted with a blue selection bar.

Node ID	Name	Type	EPG	IPV4 Addr	IPV4 Gate	IPV6 Address	IPV6 Gateway
pod-1/node-1	apic1	Out-Of-Band	default	1...	1...
pod-1/node-2	apic2	Out-Of-Band	default	1...	1...
pod-1/node-3	vapic3	Out-Of-Band	default	1...	1...
pod-1/node-101	LEAF101	Out-Of-Band	default	1...	1...
pod-1/node-102	LEAF102	Out-Of-Band	default	1...	1...
pod-1/node-201	SPINE201	Out-Of-Band	default	1...	1...
pod-1/node-202	SPINE202	Out-Of-Band	default	1...	1...

Figure 3-14 *Static Node Management Addresses View After OOB IP Configuration*

[Chapter 13](#) covers dynamic out-of-band and in-band management in more detail.

Applying a Default Contract to Out-of-Band Subnet

From a high level, contracts enable the enforcement of security and other policies to the endpoints to which the contract associates. As a fabric initialization task, administrators can assign an out-of-the-box contract called default from a tenant called common to the OOB EPG to allow all communication to and from the OOB subnet.

While assignment of contracts permitting all communication is not an ideal long-term approach, it does enable the gradual enforcement of security policies as requirements are better understood. Moreover, the application of a contract is necessary when enabling certain management protocols, such as Telnet. Also, even though it is not required to implement an OOB contract for certain features like syslog forwarding to work, it is best practice to do so.



To apply the default OOB contract to the OOB management EPG, navigate to the mgmt tenant, open the Node Management EPG folder, and select Out-of-Band EPG - default. Then, in the Provided Out-of-Band Contracts section, select the contract common/default and click Update and click Submit (see [Figure 3-15](#)).



After application of a contract on an OOB EPG, a mechanism is needed to define the subnets outside the fabric that will have open access to the ACI out-of-band IP addresses assigned to the OOB management EPG. The mechanism used for management connectivity is an external management network instance profile. Navigate to the mgmt tenant, right-click the External Management Network

Instance Profile folder, and select Create External Management Network Instance Profile. Provide a name for the object and select the default contract from the common tenant in the Consumed Out-of-Band Contracts section. Finally, enter the subnets that should be allowed to communicate with the ACI OOB EPG in the Subnets section, select Update, and then click Submit. To enable all subnets to communicate with ACI over the OOB interfaces, enter the subnet 0.0.0.0/0. Alternatively, you can enter all private IP address ranges or specific subnets assigned to administrators. [Figure 3-16](#) shows the creation of an external management network instance profile.

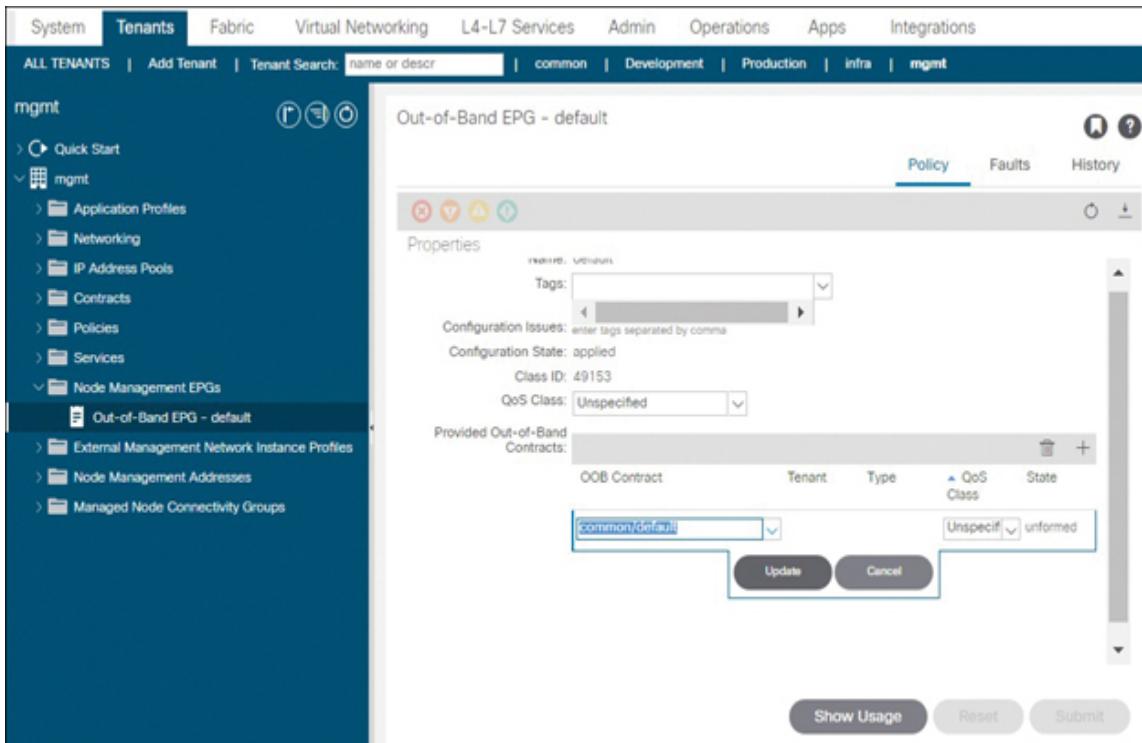


Figure 3-15 Assigning a Contract to an OOB Management EPG

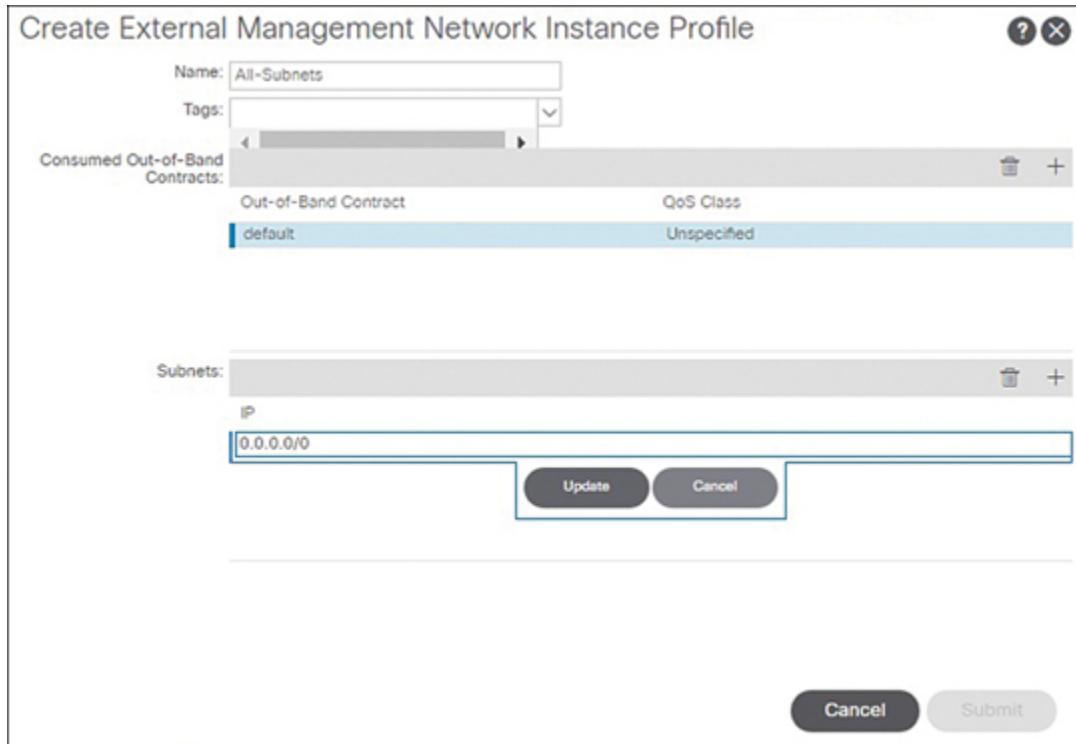


Figure 3-16 Creating an External Management Network Instance Profile

To recap, it is important to enforce contracts for access to the OOB management interface of an ACI fabric because certain configurations rely on contract enforcement. For open communication to the OOB subnets through use of contracts, take the following three steps:



Step 1. Assign static node management addresses:

Assign out-of-band addresses to all switches and APICs in the fabric and ensure that all nodes are shown in the Static Node Management Addresses view.

Step 2. Assign contract to the desired out-of-band EPG:

By default, the object called Out-of-Band

EPGs - default represents OOB subnets. Assigning a contract that allows all traffic, such as the contract named common/default, can enable open communication to OOB subnets.

Step 3. Define external management network

instance profiles and associate contracts: An external management network instance profile determines the subnets that can gain management access to ACI. Allocate the same contract applied in the previous step to the external management network instance profile you create to ensure that the contract is enforced between the external subnets you define and the ACI OOB subnets.

Upgrading an ACI Fabric

As a best practice, all nodes within an ACI fabric should operate at the same code revision. Upon purchasing ACI switches and APICs and setting up an ACI fabric, it is highly likely that components may have been shipped at different code levels. For this reason, it is common practice for engineers to upgrade ACI fabrics right after initialization.

If there are version disparities between APICs and ACI switch code, it is also possible for the APICs to flag certain switches as requiring electronic programmable logic device (EPLD) upgrades. EPLD upgrades enhance hardware functionality and resolve known issues with hardware firmware. EPLD upgrade code is sometimes slipstreamed into ACI firmware images, and therefore an EPLD upgrade may take place automatically as part of ACI fabric upgrades.

The first thing to do when upgrading a fabric is to decide on a target code. Consult the release notes for candidate target software revisions and review any associated open software defects. Also, use the APIC Upgrade/Downgrade Support

Matrix from Cisco to determine if there are any intermediary code upgrades required to reach the targeted code.

After selecting a target software revision, download the desired APIC and switch code from the Cisco website. ACI switch and APIC firmware images that can be used for upgrades have the file extensions .bin and .iso, respectively.

The ACI fabric upgrade process involves three steps:

Step 1. Download APIC and switch software images and then upload them to the APICs.

Step 2. Upgrade APICs.

Step 3. Upgrade spine and leaf switches in groups.

To upload firmware images to ACI, navigate to the Admin tab, click on Firmware, select Images, click the Tools icon, and then select Add Firmware to APIC (see [Figure 3-17](#)).



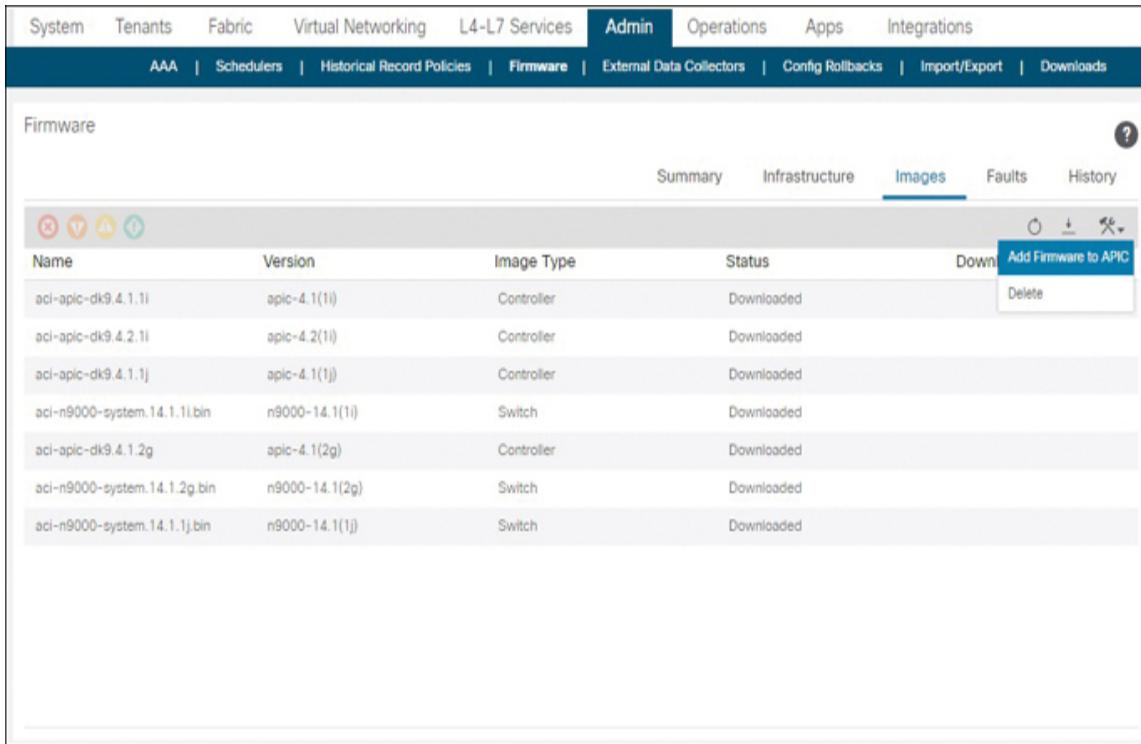


Figure 3-17 Navigating to the Add Firmware to APIC Page

In the Add Firmware to APIC page, keep Firmware Image Location set at its default value, Local, and then click Browse to select a file for upload from the local device from which the web session to the APIC has been established.

Alternatively, either HTTP or SCP (Secure Copy Protocol) can be used to download the target software code from a remote server. To download the image from a remote server, select the Remote option under Firmware Image Location and enter a name and URL for the download operation. SCP authenticates and encrypts file transfers and therefore additionally requires entry of a username and password with access to download rights on the SCP server. Instead of using a password, you can have ACI leverage SSH key data for the SCP download. [Figure 3-18](#) shows sample data for downloading a file from an SCP server using a local username and password configured on the SCP server.



Figure 3-18 Downloading Firmware from a Remote SCP Server

Once the firmware images have been uploaded to the APICs, they appear in the Images view (refer to [Figure 3-17](#)).



Unless release notes or the APIC Upgrade/Downgrade Support Matrix for a target release indicates otherwise, APICs should always be upgraded first. Navigate to the Admin menu, select Firmware, and click Infrastructure. Under the Controllers menu, click the Tools icon and select Schedule Controller Upgrade, as shown in [Figure 3-19](#).

Firmware

Ignore Compatibility Check: true
Target Firmware Version: apic-4.1(1j)
Start time: 2019-11-05 21:17:03.456+00:00

ID	Name	Role	Model	Current Firmware	Status	Upgrade Progress
1	apic1	controller	APIC-SERVER-M2	4.1(1j)	Upgraded successfully on ...	<div style="width: 100%;">100%</div>
2	apic2	controller	APIC-SERVER-M2	4.1(1j)	Upgraded successfully on ...	<div style="width: 100%;">100%</div>
3	vapi3	controller	VMware Virtual Platform	4.1(1j)	Upgraded successfully on ...	<div style="width: 100%;">100%</div>

Figure 3-19 Navigating to the Schedule Controller Upgrade Page



The Schedule Controller Upgrade page opens. ACI advises against the upgrade if any critical or major faults exist in the fabric. These faults point to important problems in the fabric and can lead to traffic disruption during or after the upgrade. Engineers are responsible for fully understanding the caveats associated with active faults within a fabric. Do not upgrade a fabric when there are doubts about the implications of a given fault. After resolving any critical and major faults, select the target firmware version, define the upgrade mode via the Upgrade Start Time field (that is, whether the upgrade should begin right away or at a specified time in the future), and then click Submit to confirm the selected APIC upgrade schedule. During APIC

upgrades, users lose management access to the APICs and need to reconnect.

Figure 3-20 shows how to kick off an immediate upgrade by selecting Upgrade Now and clicking Submit.

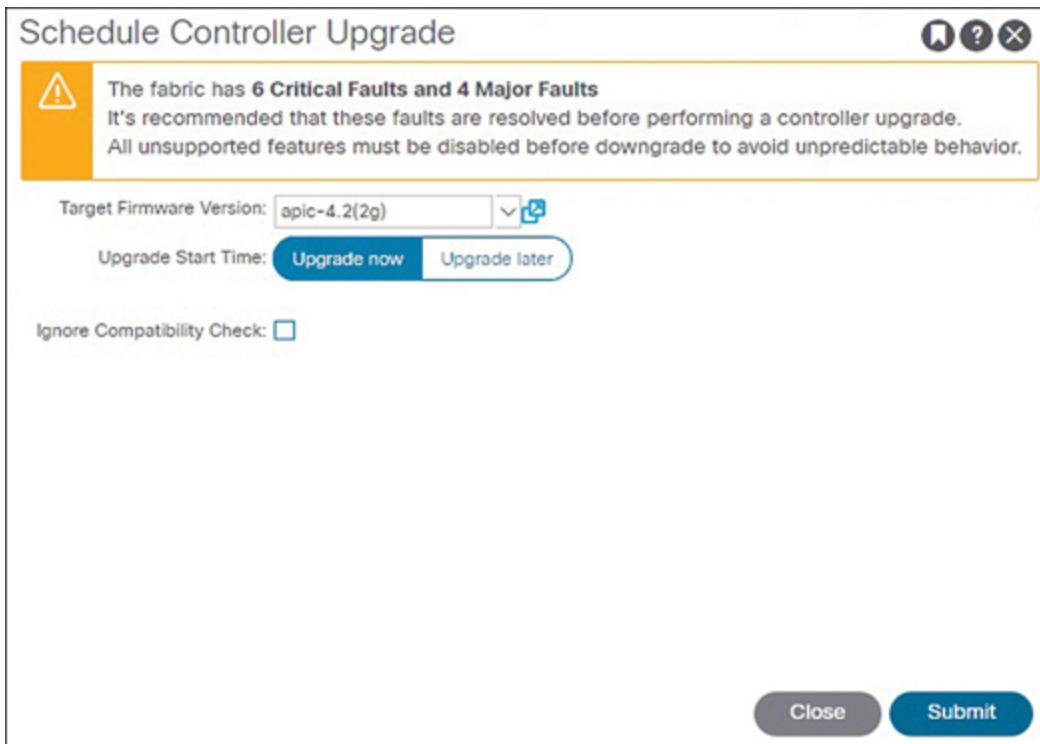


Figure 3-20 Schedule Controller Upgrade Page

By default, ACI verifies whether the upgrade path from the currently running version of the system to a specific newer version is supported. If, for any reason, ACI does not allow an upgrade due to the compatibility checks, and this is determined to be a false positive or if you wish to proceed with the upgrade anyway, you can enable the Ignore Compatibility Checks setting shown in [Figure 3-20](#).

Following completion of any APIC upgrades, switch upgrades can begin. Cisco ACI uses the concept of upgrade groups to execute a group of switch upgrades consecutively. The idea behind upgrade groups is that if all servers have been dual

connected to an odd and even switch, then an upgrade group consisting of all odd leaf switches should not lead to server traffic disruption as long as the even leaf upgrades do not happen until all odd leaf switches have fully recovered. Furthermore, if only half of all available spine switches are upgraded simultaneously and an even number of spines have been deployed, then there is little likelihood of unexpected traffic disruption.

In a hypothetical upgrade group setup, a fabric could be divided into the following four groups:

- Odd spine switches
- Even spine switches
- Odd leaf switches
- Even leaf switches

Note

Cisco only provides general guidance on configuration of upgrade groups. To maintain connectivity in a production environment, Cisco suggests that administrators define a *minimum of two* upgrade groups and upgrade one group at a time. Performing a minimally disruptive upgrade with two upgrade groups requires an administrator to group and upgrade a set of spine switches and leaf switches together. Most environments, however, tend to separate switches out into four or more upgrade groups to reduce the risk and extent of downtime if, for any reason, something goes wrong.

Key Topic

To configure an upgrade group, navigate to the Admin menu, select Firmware, click Infrastructure, and then select Nodes. Open the Tools menu and select Schedule Node Upgrade, as shown in [Figure 3-21](#).

The screenshot shows the Cisco Fabric Manager interface with the following navigation path: System > Admin > Firmware > Infrastructure > Nodes. The main content area displays a table of nodes with columns: ID, Name, Role, Model, Current Firmware, Upgrade Group, Status, and Upgrade Progress. The nodes listed are: Pod1/101 (LEAF101, leaf, N9K-C93180YC-EX, n9000-14.1(1)), Pod1/102 (LEAF102, leaf, N9K-C93180YC-EX, n9000-14.1(1)), Pod1/201 (SPINE201, spine, N9K-C9336PQ, n9000-14.1(1)), and Pod1/202 (SPINE202, spine, N9K-C9332C, n9000-14.1(1)). At the bottom right of the table, there is a blue button labeled "Schedule Node Upgrade". Below the table, there are "Reset" and "Submit" buttons.

ID	Name	Role	Model	Current Firmware	Upgrade Group	Status	Upgrade Progress
Pod1/101	LEAF101	leaf	N9K-C93180YC-EX	n9000-14.1(1)		Not Scheduled	
Pod1/102	LEAF102	leaf	N9K-C93180YC-EX	n9000-14.1(1)		Not Scheduled	
Pod1/201	SPINE201	spine	N9K-C9336PQ	n9000-14.1(1)		Not Scheduled	
Pod1/202	SPINE202	spine	N9K-C9332C	n9000-14.1(1)		Not Scheduled	

Figure 3-21 Navigating to Schedule Node Upgrade

Key Topic

In the Schedule Node Upgrade window, select New in the Upgrade Group field, choose a target firmware version, select an upgrade start time, and then select the switches that should be placed in the upgrade group by clicking the + sign in the All Nodes view. Nodes can be selected from a range based on node IDs or manually one by one. Finally,

click Submit to execute the upgrade group creation and confirm scheduling of the upgrade of all switches that are members of this new upgrade group. [Figure 3-22](#) shows the creation of an upgrade group called ODD-SPINES and scheduling of the upgrade of relevant nodes to take place right away. The completion of upgrades of all switches in an upgrade group can take anywhere from 12 to 30 minutes.

The Graceful Maintenance option ensures that the switches in the upgrade group are put into maintenance mode and removed from the server traffic forwarding path before the upgrade begins. The Run Mode option determines whether ACI will proceed with any subsequently triggered upgrades that may be in queue if a failure of the current upgrade group takes place. The default value for this parameter is Pause upon Upgrade Failure, and in most cases it is best not to modify this setting from its default.

Schedule Node Upgrade

Group Type: Switch vPod

Upgrade Group: Existing New

Upgrade Group Name: ODD-SPINES

Manual Silent Roll:

Package Upgrade:

Target Firmware Version: n9000-14.2(2g)

Ignore Compatibility Check:

Graceful Maintenance:

Run Mode: Do not pause on failure and do not wait on cluster health Pause upon upgrade failure

Upgrade Start Time: Now

All Nodes

ID	Name	Role	Model	Current Firmware	Status
Pod1/201	SPINE201	spine	N9K-C9336PO	n9000-14.1(1)	Not Scheduled

Figure 3-22 Creating an Upgrade Group and Scheduling Node Upgrades

One of the checkboxes shown but disabled in [Figure 3-22](#) is Manual Silent Roll Package Upgrade. A silent roll package upgrade is an internal package upgrade for an ACI switch hardware SDK, drivers, or other internal components without an upgrade of the entire ACI switch software operating system. Typically, you do not need to perform a silent roll upgrade because upgrading the ACI switch operating system takes care of internal packages as well. Each upgrade group can be dedicated to either silent roll package upgrades or firmware upgrades but not both. Thus, the selection of a firmware code revision from the Target Firmware Version pull-down disables the Manual Silent Roll Package checkbox.

The triggering of an upgrade group places all switches in the specified upgrade group into queue for upgrades to the targeted firmware version. If upgrades for a group of nodes have been scheduled to start right away and no prior upgrade group is undergoing upgrades, the node upgrades can begin right away. Otherwise, the nodes are placed into queue for upgrades of previous upgrade groups to complete (see [Figure 3-23](#)). As indicated in [Figure 3-23](#), the EVEN-SPINES group needs to wait its turn and allow upgrades of nodes in the ODD-LEAFS group to finish first.

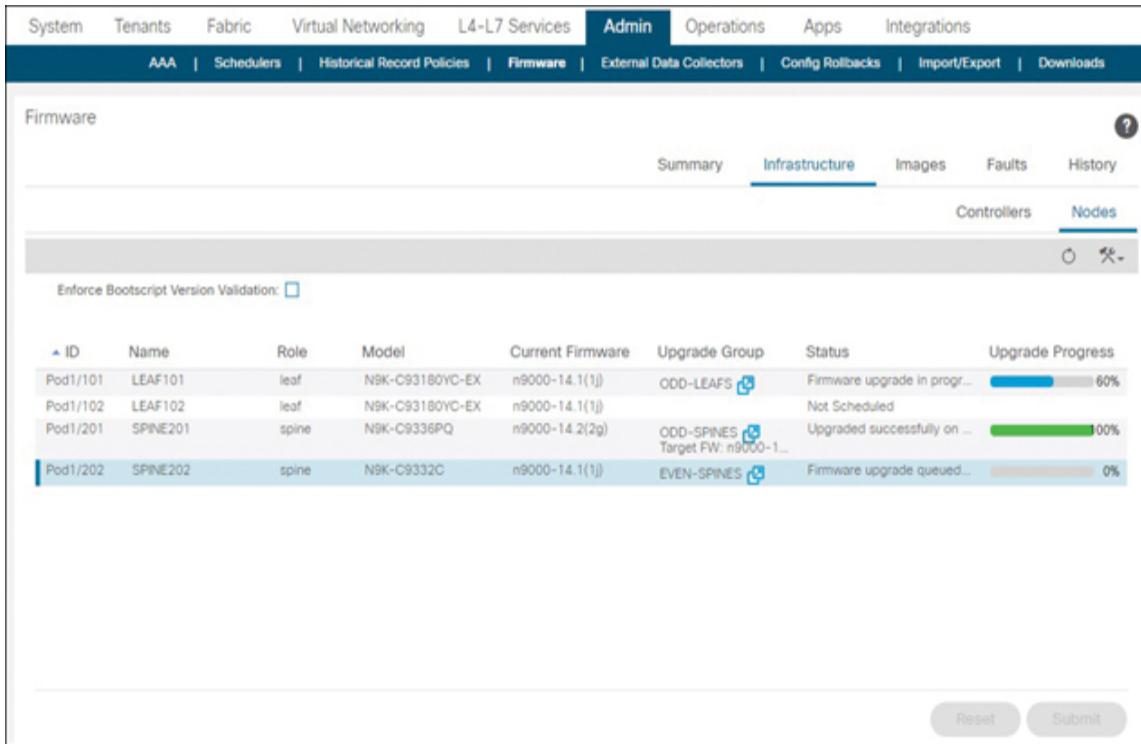


Figure 3-23 An Upgrade Group Placed into Queue Due to Ongoing Upgrades

Cisco recommends that ACI switches be divided into two or more upgrade groups. No more than 20 switches can be placed into a single upgrade group. Switches should be placed into upgrade groups to ensure maximum redundancy. If, for example, all spine switches are placed into a single upgrade group, major traffic disruption should be expected.

Once an upgrade group has been created, the grouping can be reused for subsequent fabric upgrades. [Figure 3-24](#) shows how the selection of Existing in the Upgrade Group field allows administrators to reuse previously created upgrade group settings and trigger new upgrades simply by modifying the target firmware revision.

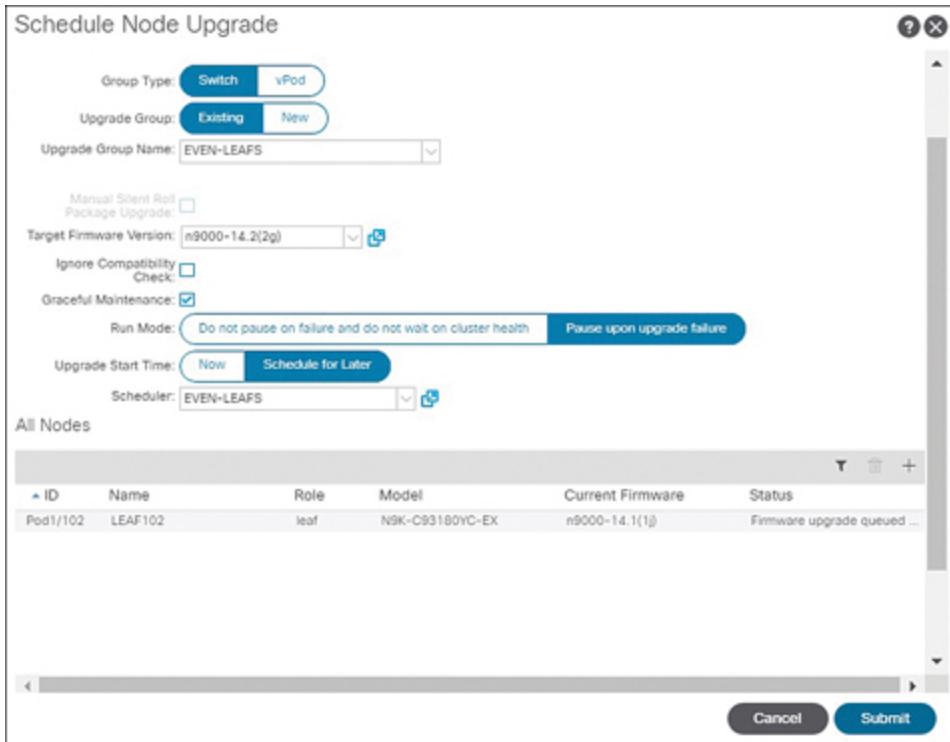


Figure 3-24 Reusing a Previously Created Upgrade Group for Subsequent Upgrades

Understanding Schedulers



An administrator can create a **scheduler** to specify a window of time for ACI to execute operations such as switch upgrades and configuration backups. Schedulers can be triggered on a one-time-only basis or can recur on a regular basis.

When an administrator creates an upgrade group, ACI automatically generates a scheduler object with the same name as the group.

In Figure 3-24 in the previous section, Schedule for Later has been selected for the Upgrade Start Time parameter, which in the installed APIC code version defaults to a scheduler with the equivalent name as the upgrade group name. The administrator can edit the selected scheduler by clicking on the blue link displayed in front of it. Figure 3-25 shows the Trigger Scheduler window, from which a one-time schedule can be implemented by hovering on the + sign and clicking Create.

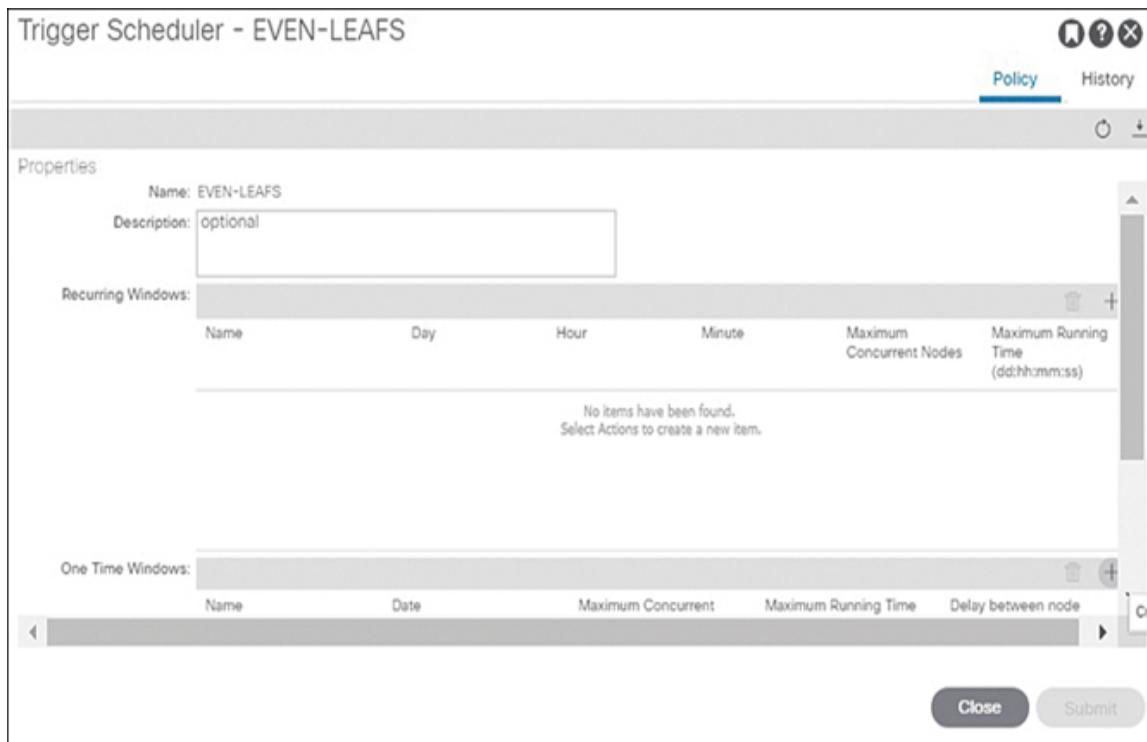


Figure 3-25 Creating a One-Time Trigger for a Scheduler

Figure 3-26 demonstrates the selection of a one-time window trigger, which involves the selection of a window name, the desired date and time, and the maximum number of nodes to upgrade simultaneously.

Create One Time Window Trigger

Name:

Date: Format: YYYY-MM-DD HH.MM.SS AM/PM

Maximum Concurrent Nodes:
Eg: 0, unlimited, defaultValue or in the range [0 - 65535]

Maximum Running Time (dd:hh:mm:ss):

Figure 3-26 Parameters Needed for Adding a One-Time Window Trigger to a Scheduler

Enabling Automatic Upgrades of New Switches

Earlier in this chapter, we mentioned that APICs can force new switches to undergo upgrades to a certain firmware version prior to moving them into an active state.



The code version to which new switches should be upgraded needs to be selected using the Default Firmware Version setting. This setting, however, may be unavailable in certain APIC code versions by default. [Figure 3-27](#) shows that after the Enforce Bootscript Version Validation setting is enabled,

an administrator can then select a value for the Default Firmware Version setting.

The screenshot shows the Cisco Application Centric Infrastructure (ACI) Admin interface. The top navigation bar includes links for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin (which is selected), Operations, Apps, and Integrations. Below the Admin link, there are tabs for AAA, Schedulers, Historical Record Policies, Firmware (selected), External Data Collectors, Config Rollbacks, Import/Export, and Downloads. The main content area is titled "Firmware" and has tabs for Summary, Infrastructure (selected), Images, Faults, and History. Under Infrastructure, it shows "Controllers" and "Nodes". A checkbox for "Enforce Bootscript Version Validation" is checked, and a dropdown menu shows the "Default Firmware Version" as "m9000-14.2(3)". A table lists four nodes: Pod1/101 (Leaf101, leaf, N9K-C93180YC-EX, m9000-14.2(3)), Pod1/102 (Leaf102, leaf, N9K-C93180YC-EX, m9000-14.2(3)), Pod1/201 (Spine201, spine, N9K-C9336PQ, m9000-14.2(3)), and Pod1/202 (Spine202, spine, N9K-C9332C, m9000-14.2(3)). All nodes show "Upgraded successfully on..." and 100% Upgrade Progress. At the bottom right are "Reset" and "Submit" buttons.

ID	Name	Role	Model	Current Firmware	Upgrade Group	Status	Upgrade Progress
Pod1/101	Leaf101	leaf	N9K-C93180YC-EX	m9000-14.2(3)	ODD-LEAF	Upgraded successfully on... Target FW: m9000-14.2(3)	100%
Pod1/102	Leaf102	leaf	N9K-C93180YC-EX	m9000-14.2(3)	EVEN-LEAFS	Upgraded successfully on... Target FW: m9000-14.2(3)	100%
Pod1/201	Spine201	spine	N9K-C9336PQ	m9000-14.2(3)	ODD-SPINES	Upgraded successfully on... Target FW: m9000-14.2(3)	100%
Pod1/202	Spine202	spine	N9K-C9332C	m9000-14.2(3)	EVEN-SPINES	Upgraded successfully on... Target FW: m9000-14.2(3)	100%

Figure 3-27 Selecting the Default Firmware Version

To execute the change, an administrator needs to click Submit. ACI then requests confirmation of the change by using an alert like the one shown in [Figure 3-28](#).

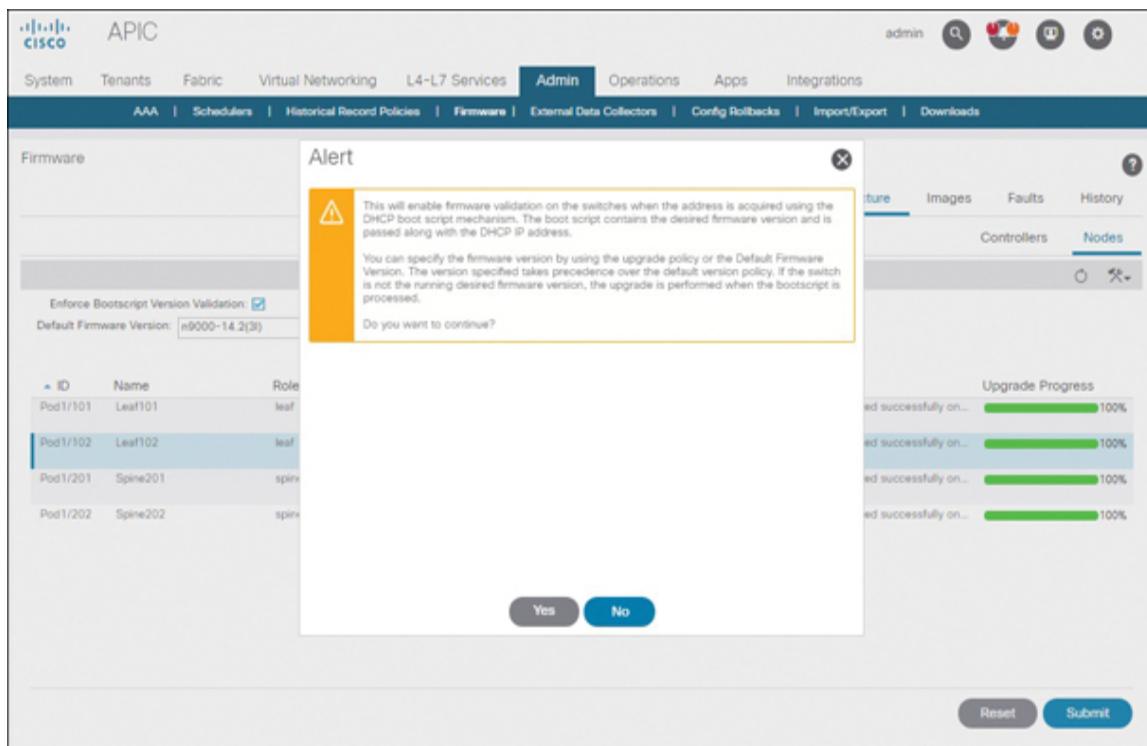


Figure 3-28 *Confirming Enforcement of Bootscript Version Validation*

From the alert message, it is clear that the code version selected for Default Firmware Version is indeed what is passed along to any new switches as part of the boot process. The alert message also clarifies that any switches whose node IDs have been added to an upgrade group will not be bound to the bootscript version requirements, as manual configuration supersedes the Default Firmware Version setting. Click Yes to confirm.

Understanding Backups and Restores in ACI

ACI allows both scheduled backups and on-demand backups of user configurations. The act of making a backup is referred to as a *configuration export*. Restoring ACI

configurations from a backup is referred to as a *configuration import*.

ACI also enables recovery of the fabric configuration to a previous known good state. This process, called *configuration rollback*, is very useful when backing out of a change window is deemed necessary. For configuration rollback to a specific point in time (for example, prior to a change window), it is important for administrators to have taken a snapshot of the fabric configuration at the specified time. Snapshots are stored locally on the APICs.

In addition to snapshots, ACI can export configurations to remote FTP, SFTP, or SCP servers.

Note

For rapid rollback of configurations, it is best to take very regular configuration snapshots. To ease disaster recovery, administrators are also advised to retain two or more remote copies of recent backups at all times. These should be stored in easily accessible locations outside the local ACI fabric and potentially offsite. To automate backups, administrators can tie ACI backup operations to schedulers.

When performing a configuration import, ACI wants to know the desired import type and import mode. Import Type can be either set to Merge or Replace. As indicated in [Table 3-5](#), the Import Type setting primarily determines what happens when the configuration being imported conflicts with the current configuration.



Table 3-5 Import Types

Import Definition Type	
Merge	The import operation combines the configuration in the backup file with the current configuration.
Replace	The import operation overwrites the current configuration with the configuration imported from the backup file.

The options for the Import Mode parameter are Best Effort and Atomic. The Import Mode parameter primarily determines what happens when configuration errors are identified in the imported settings. [Table 3-6](#) describes the Import Mode options.

Table 3-6 Import Mode

Import Mode Definition
Best Effort

B	Each shard is imported, but if there are objects within a shard that are invalid, these objects are ignored and not imported. If the version of the configuration being imported is incompatible with the current system, shards that can be imported are imported, and all other shards are ignored.
At	The import operation is attempted for each shard, but if a shard has any invalid configuration, the shard is ignored and not imported. Also, if the version of the configuration being imported is incompatible with the current system, the import operation terminates.

An import operation configured for atomic replacement, therefore, attempts to import all configurations from the backup and attempts to overwrite all settings to those specified in the backup file. Where a backup file may be used to import configurations to a different fabric, a best-effort merge operation may be a more suitable fit.



Note that when an administrator selects Replace as the import type in the ACI GUI, the administrator no longer has the option to choose an import mode. This is because the import mode is automatically set at the default value Atomic to prevent a situation in which an import type Replace and an import mode Best Effort might break the fabric.

Another important aspect of backup and restore operations is whether secure properties are exported into backup files or processed from imported files. Secure properties are parameters such as SNMP or SFTP credentials or credentials used for integration with third-party appliances. For ACI to include these parameters in backup files and process secure properties included in a backup, the fabric needs to be configured with global AES encryption settings.

Making On-Demand Backups in ACI

To take an on-demand backup of an ACI fabric, navigate to the Admin tab, select Import/Export, open the Export Policies folder, right-click Configuration, and select Create Configuration Export Policy, as shown in [Figure 3-29](#).

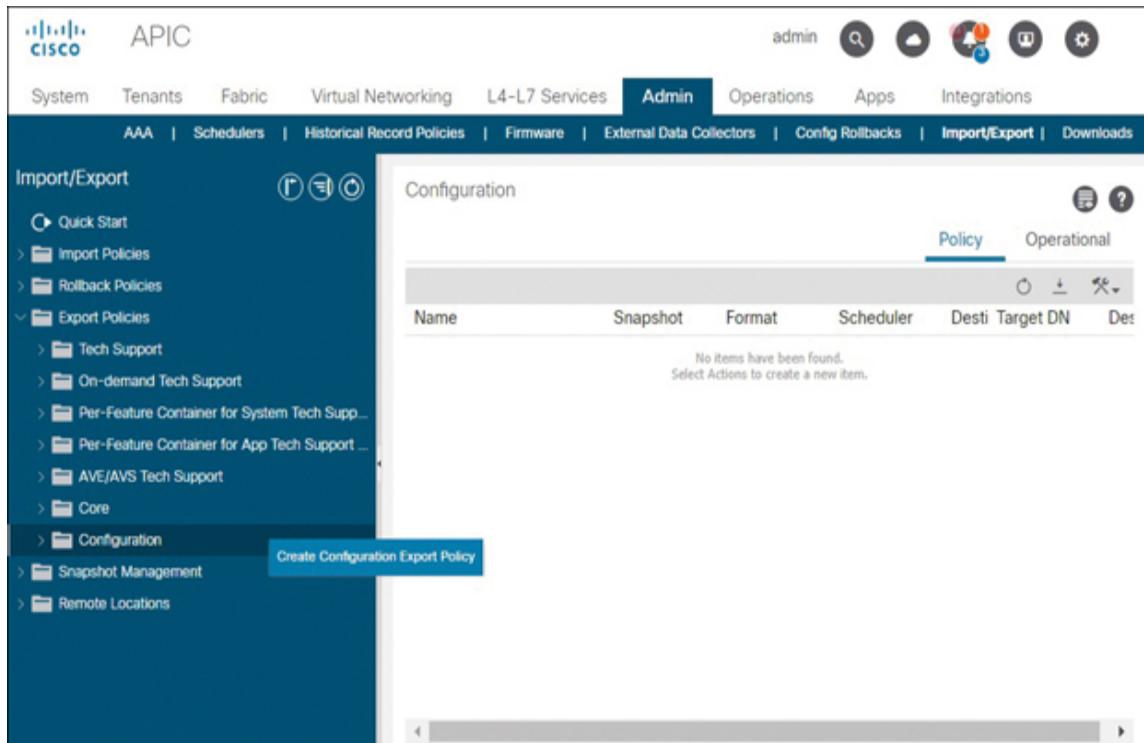


Figure 3-29 Navigating to the Configuration Import Wizard

In the Create Configuration Export Policy wizard, select a name, select whether the backup file should conform with JSON or XML format, indicate that a backup should be generated right after clicking Submit by toggling Start Now to Yes, and select to create a new remote server destination by right-clicking Export Destination and selecting Create Remote Location (see [Figure 3-30](#)).

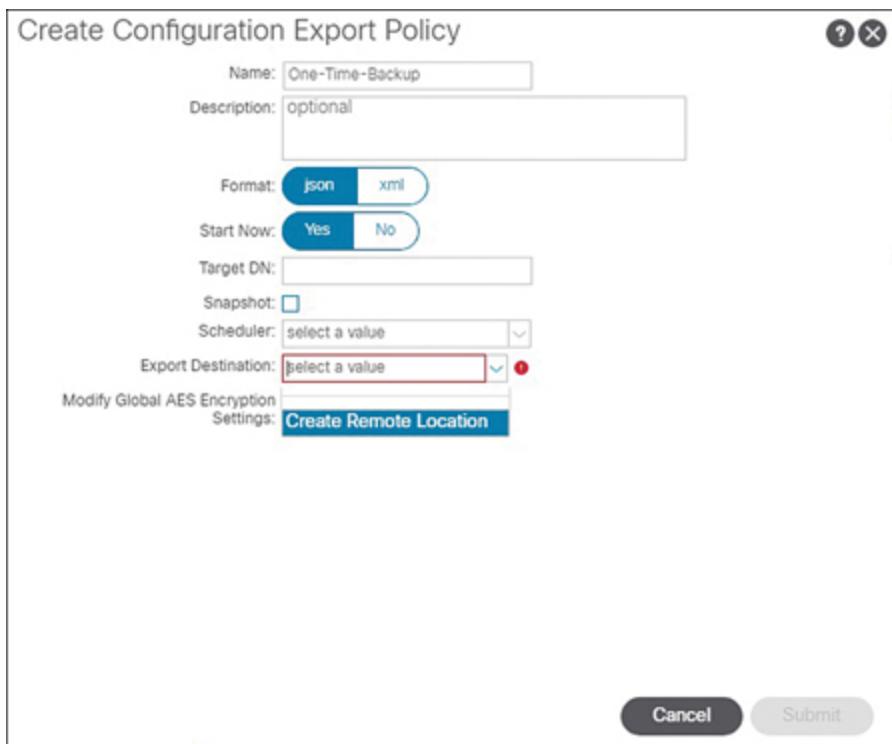


Figure 3-30 The Create Configuration Export Policy Wizard

[Figure 3-31](#) shows the Create Remote Location wizard. Enter the details pertinent to the remote server on which ACI should copy the file, and then click Submit.

Create Remote Location

Name: SFTP-Server1

Description: optional

Host Name (or IP Address): 10.233.48.11

Protocol: ftp scp sftp

Remote Path: /

Remote Port: 22

Username: aci-backup-user

Authentication Type: Use Password Use SSH Public/Private Files

Password:

Confirm Password:

Management EPG: default (Out-of-Band)

Cancel

The screenshot displays the 'Create Remote Location' configuration window. At the top, there are buttons for help (?) and close (X). The main area contains fields for 'Name' (SFTP-Server1), 'Description' (optional), 'Host Name (or IP Address)' (10.233.48.11), 'Protocol' (with 'sftp' selected), 'Remote Path' (/), 'Remote Port' (22), 'Username' (aci-backup-user), 'Authentication Type' (with 'Use Password' selected), 'Password' (redacted), 'Confirm Password' (redacted), and 'Management EPG' (default (Out-of-Band)). At the bottom are 'Cancel' and 'Submit' buttons.

Figure 3-31 The Create Remote Location Wizard

Finally, back in the Create Configuration Export Policy wizard, update the global AES encryption settings, if desired. Click the Modify Global AES Encryption Settings checkbox to enable encryption of secure properties, as shown in [Figure 3-32](#).

Create Configuration Export Policy

Name: One-Time-Backup

Description: optional

Format: json xml

Start Now: Yes No

Target DN:

Snapshot:

Scheduler: select a value

Export Destination: SFTP-Server1

Modify Global AES Encryption Settings: **Disabled**

To export hashed secure properties (**passwords and certificates**), AES encryption must be configured & enabled. While encryption is not enabled, any secure fields will not be exported.
In this case, re-importing the configuration would require all secure properties to be re-configured.

Figure 3-32 Navigating to Global AES Encryption from the Export Window

In the Global AES Encryption Settings for All Configuration Import and Export page, shown in [Figure 3-33](#), select the Enable Encryption checkbox and then enter the passphrase for encryption. The passphrase needs to be between 16 to 32 characters.

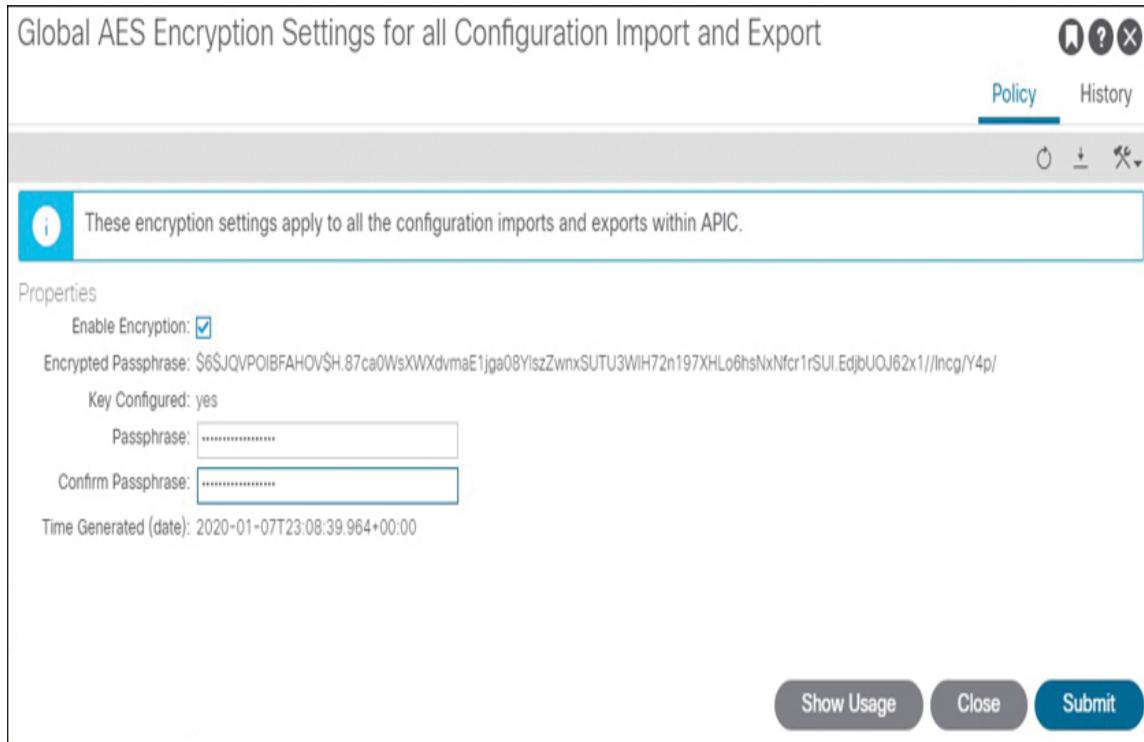


Figure 3-33 Entering Encryption Settings in the Wizard

Click Submit to return to the Create Configuration Export Policy wizard. With encryption enabled, secure properties will also be included in backup files.

Finally, click Submit to execute the configuration backup.

Note that one of the options available when making configuration backups is to specify the target DN field. This field limits the backup to a specific portion of the ACI object hierarchy. When this field is not populated, the policy universe and all subtrees are captured in the backup file. [Chapter 4, “Exploring ACI,”](#) introduces the ACI object hierarchy in detail.

Making Scheduled Backups in ACI

Scheduled backups are very similar to one-time backups. However, a scheduled backup also includes a reference to a

scheduler object. For instance, an administrator who wants the entire fabric to be backed up every four hours could enter settings similar to the ones shown in [Figure 3-34](#).

The screenshot shows a configuration dialog titled "Create Configuration Export Policy". The "Name" field is set to "Automated-Backups-4-Hours". The "Format" section has "json" selected. The "Start Now" button is set to "Yes". The "Target DN" and "Snapshot" fields are empty. The "Scheduler" dropdown is set to "Every-4-Hours" and the "Export Destination" dropdown is set to "SFTP-Server1". A "Modify Global AES Encryption Settings" section is present with an "Enabled" checkbox checked. At the bottom are "Cancel" and "Submit" buttons.

Figure 3-34 Configuring Automated Backups Using a Recurring Schedule

A scheduler that enables backups every four hours would need six entries, each configured for execution on a specific hour of day, four hours apart (see [Figure 3-35](#)).

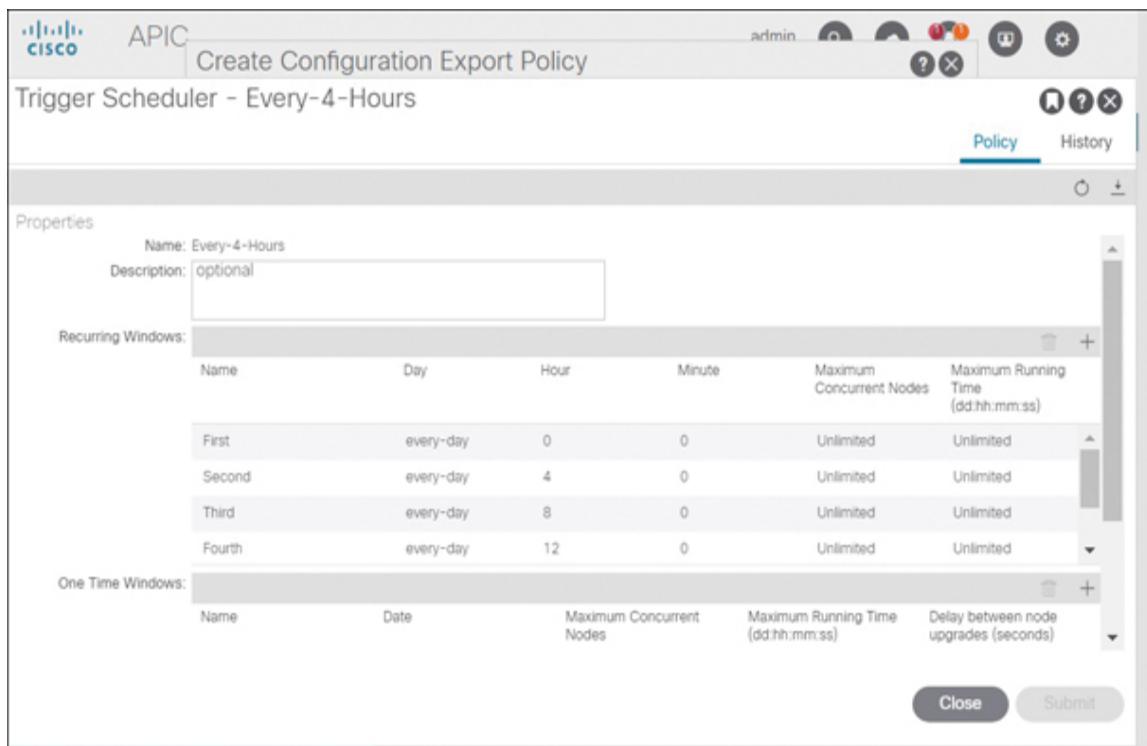


Figure 3-35 A Scheduler That Triggers an Action Every Four Hours

Taking Configuration Snapshots in ACI

In addition to backing up configurations to remote locations, ACI allows users to take a snapshot of the configuration for local storage on the APICs. This can be done by enabling the Snapshot checkbox in the Create Configuration Export Policy wizard. [Figure 3-36](#) shows that when the Snapshot checkbox is enabled, ACI removes the option to export backups to remote destinations.

Create Configuration Export Policy

Name:

Description:

Format: json xml

Start Now: Yes No

Target DN:

Snapshot:

Scheduler: Every-4-Hours

Modify Global AES Encryption
Settings: Enabled

Figure 3-36 Creating Snapshots of ACI Configurations on a Recurring Basis

Importing Configuration Backups from Remote Servers

To restore a configuration from a backup that resides on a remote server, navigate to the Admin tab, select Import/Export, drill into the Import Policies folder, right-click on Configuration, and then select Create Configuration Import Policy (see [Figure 3-37](#)).

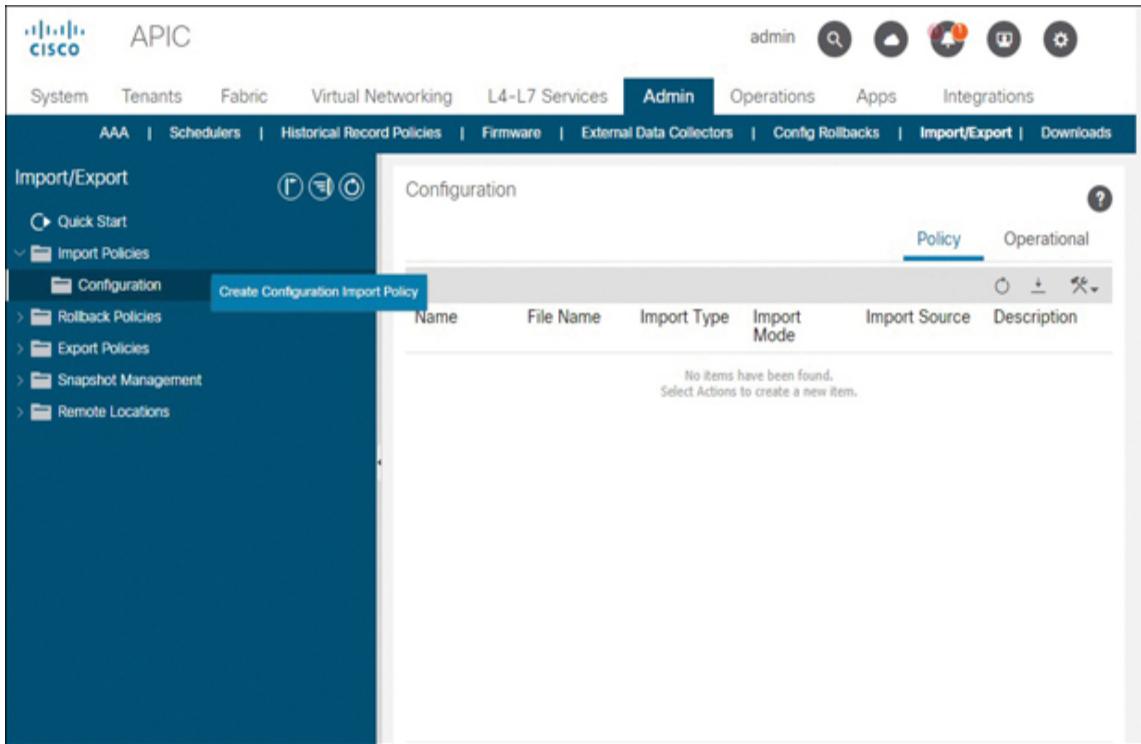


Figure 3-37 Navigating to the Create Configuration Import Policy Wizard

In the Create Configuration Import Policy wizard, enter a name for the import operation, enter details of the backup filename, select the import type and import mode, select the encryption settings, enter whether the process should start right away, and enter the remote destination from which the backup file should be downloaded. [Figure 3-38](#) shows a sample import operation using Atomic Replace to restore all configuration to that specified in the backup file. Remember that when Import Type is set to Replace, Import Mode cannot be set to Best Effort.

Create Configuration Import Policy

Name:

Description:

File Name:
File name ending with .tar.gz

Import Type:

Fail Import if secure fields cannot be decrypted:

Modify Global AES Encryption Settings:

Start Now:

Snapshot:

Import Source:

Figure 3-38 Restoring the Configuration from a Backup Residing on an External Server

Once executed, the status of the import operation can be verified in the Operational tab of the newly created object, as shown in [Figure 3-39](#).

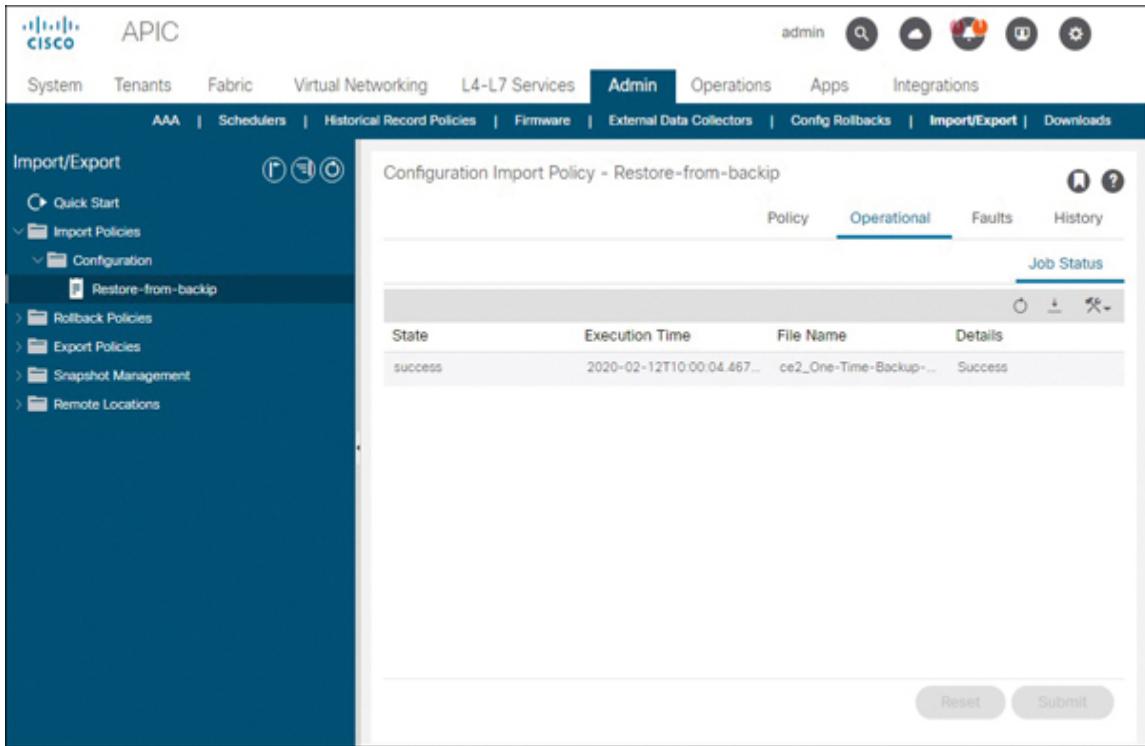


Figure 3-39 Verifying the Status of an Import Operation

In instances in which secure properties are not encrypted or a test of a backup and restore operation is desired, use of a configuration merge may be more desirable. [Figure 3-40](#) shows that if Import Type is set to Merge, Import Mode can be set to Best Effort.

Create Configuration Import Policy

Name: Merge-with-config

Description: optional

File Name: ce2_One-Time-Backup-2020-02-12T09-51-38.tar.gz
File name ending with .tar.gz

Import Type: Merge Replace

Import Mode: Atomic Best Effort

Fail Import if secure fields cannot be decrypted:

Modify Global AES Encryption Settings: Enabled

Start Now: Yes No

Snapshot:

Import Source: SFTP-Server1

Cancel Submit

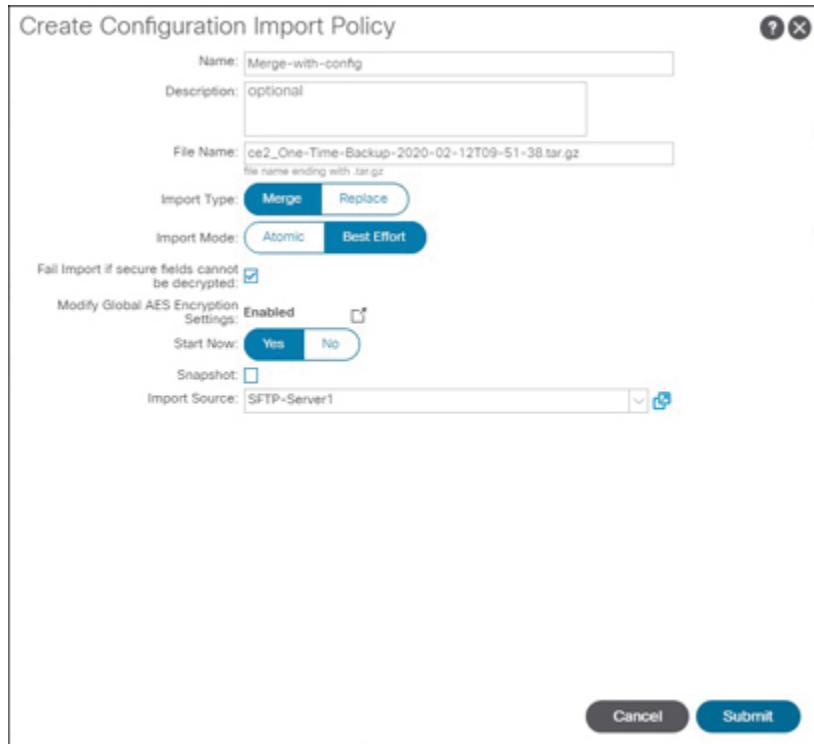


Figure 3-40 Merging a Configuration Backup with Current Configurations

Executing Configuration Rollbacks

When a misconfiguration occurs and there is a need to restore back to an earlier configuration, you can execute a configuration rollback. To do so, navigate to the Admin tab and select Config Rollback. Then select the configuration to which ACI should roll back from the list and select Rollback to This Configuration, as shown in Figure 3-41.

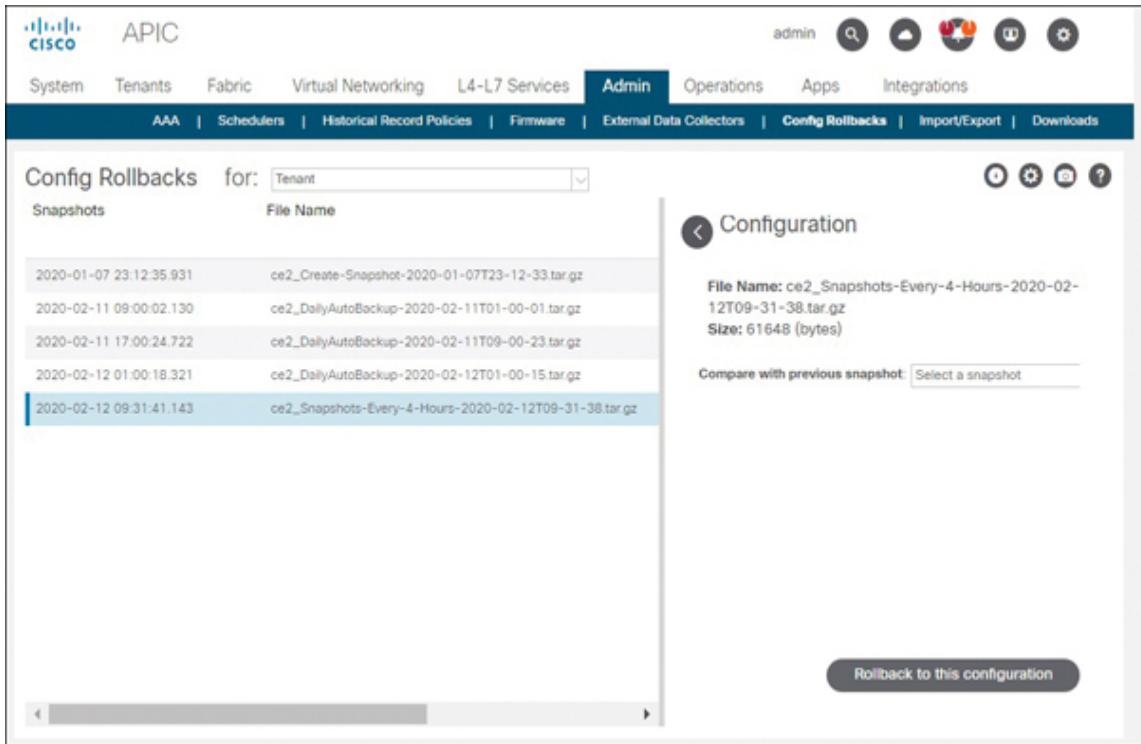


Figure 3-41 Executing a Configuration Rollback

Note that one of the beauties of configuration rollbacks and backups in ACI in general is that configurations can be backed up and restored fabricwide, for a single tenant, or for any specific portion of the ACI fabric object hierarchy.

ACI also simplifies pre-change snapshot creations by allowing users to take snapshots directly from within the Config Rollback page.

In instances in which a user does not know which snapshot is the most suitable to revert to, ACI can be directed to compare the contents of snapshots with one another and log differences between the selected snapshots.

Pod Policy Basics

Key Topic

All switches in ACI reside in a pod. This is true whether ACI Multi-Pod has been deployed or not. In single-pod deployments, ACI places all switches under a pod profile called default. Because each pod runs different control plane protocol instances, administrators need to have a way to modify configurations that apply to pods. Another reason for the need to tweak pod policies is that different pods may be in different locations and therefore may need to synchronize to different NTP servers or talk to different SNMP servers.

Key Topic

A **pod profile** specifies date and time, podwide SNMP, COOP settings, and IS-IS and Border Gateway Protocol (BGP) route reflector policies for one or more pods. Pod profiles map pod policy groups to pods by using pod selectors:

- A **pod policy group** is a group of individual protocol settings that are collectively applied to a pod.
- A **pod selector** is an object that references the pod IDs to which pod policies apply. Pod policy groups get bound to a pod through a pod selector.

[Figure 3-42](#) illustrates how the default pod profile (shown as Pod Profile - default) in an ACI deployment binds a pod policy group called Pod-PolGrp to all pods within the fabric.

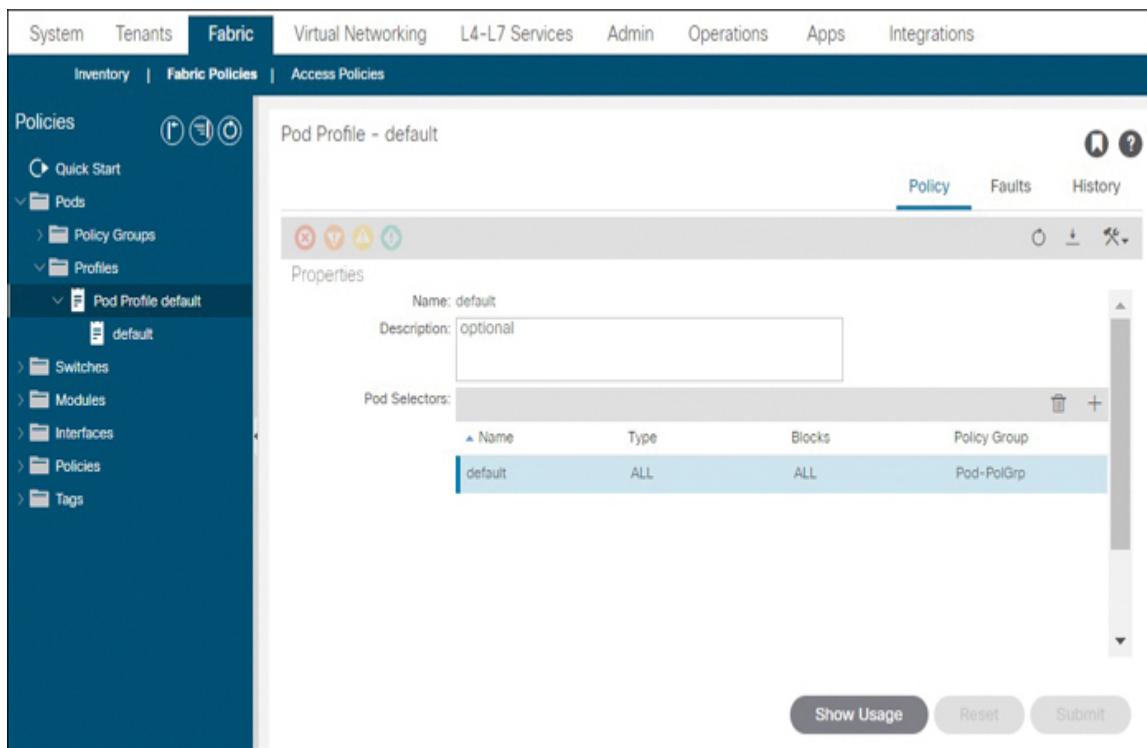


Figure 3-42 Pod Profiles, Pod Policy Groups, and Pod Selectors

Configuring Network Time Protocol (NTP) Synchronization

One of the day 0 tasks that may require changes to the default pod profile settings is NTP synchronization. Since multiple data centers may house pods from a single ACI Multi-Pod deployment, each pod may need to synchronize to different NTP servers. This is why NTP synchronization needs to be configured at the pod level.

To modify the list of NTP servers a pod points to, navigate to Fabric, select Fabric Policies, open the Pods folder, double-click Profiles, double-click the pod profile for the pod in question, select the relevant pod policy group, and click on the blue icon in front of the pod policy group to open the pod policy group applicable to the pod. Pod policy groups

are also called fabric policy groups in several spots in the ACI GUI (see [Figure 3-43](#)).

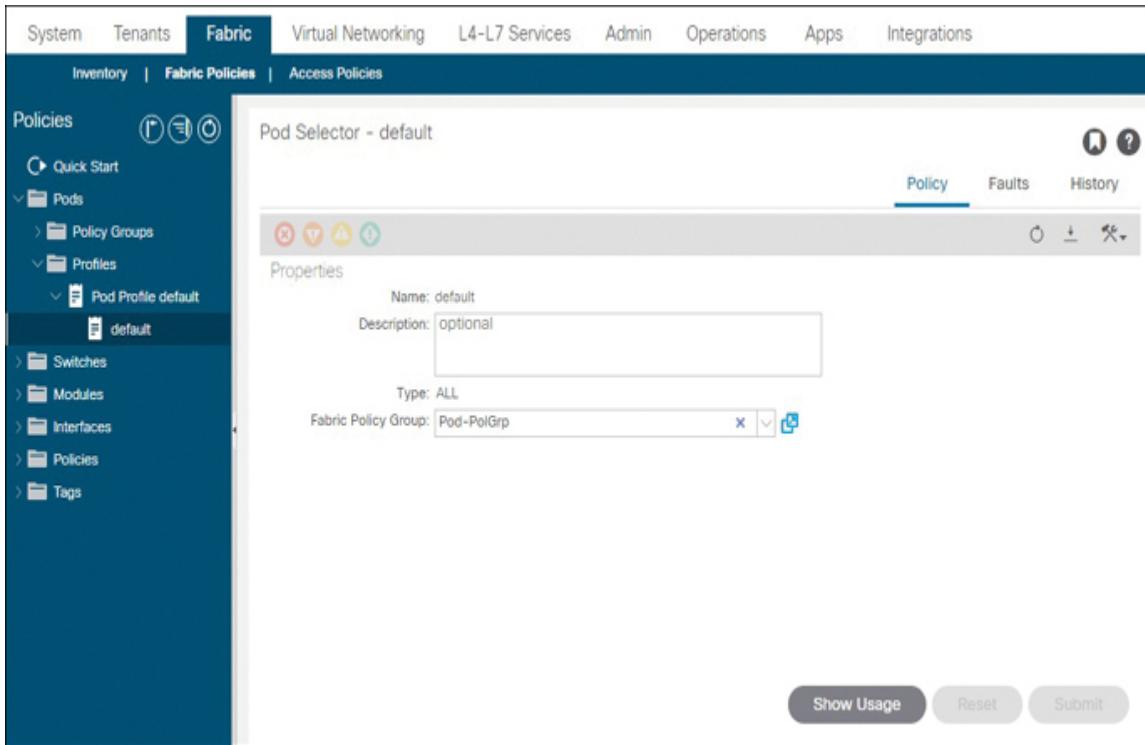


Figure 3-43 Opening the Pod Policy Group for the Relevant Pod

In the Pod Policy Group view, validate the name of the date and time policy currently applicable to the pod in question. According to [Figure 3-44](#), the date and time policy that ACI resolves for all pods in a particular deployment is a date and time policy called default.

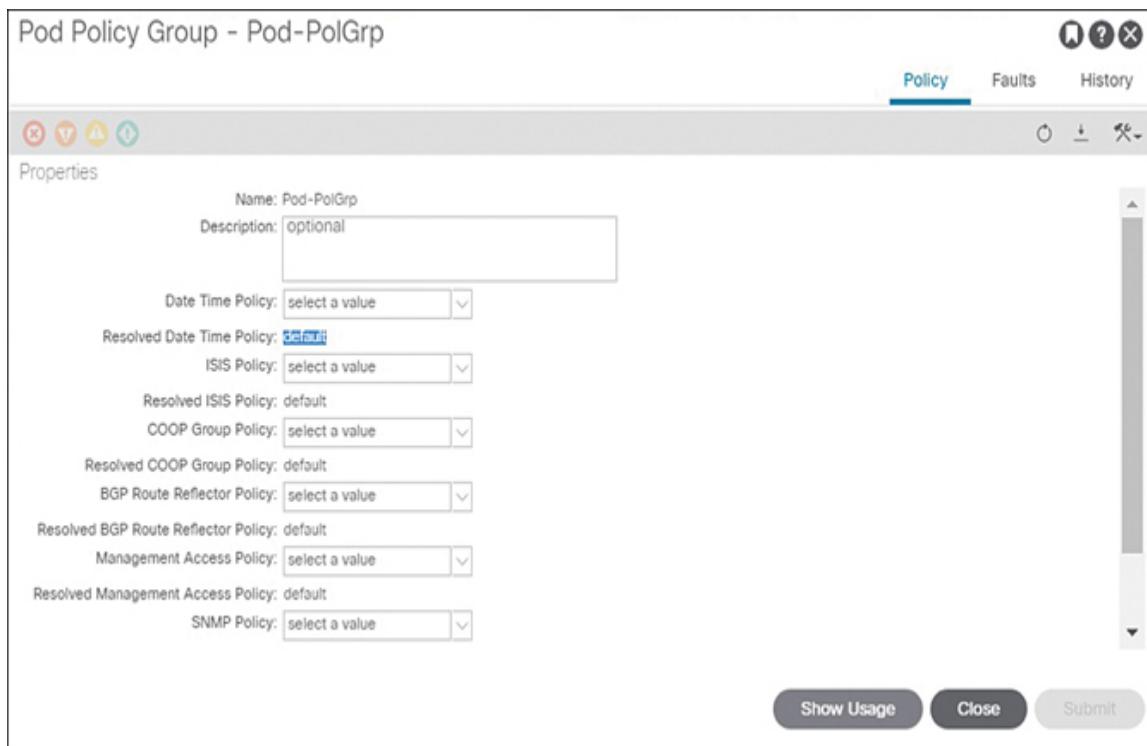


Figure 3-44 Verifying the Date and Time Policy Applied to a Pod

After identifying the date and time policy object that has been applied to the pod of interest, an administrator can either modify the applicable date and time policy or create and apply a new policy object. [Figure 3-45](#) shows how the administrator can create a new date and time policy from the Pod Policy Group view.

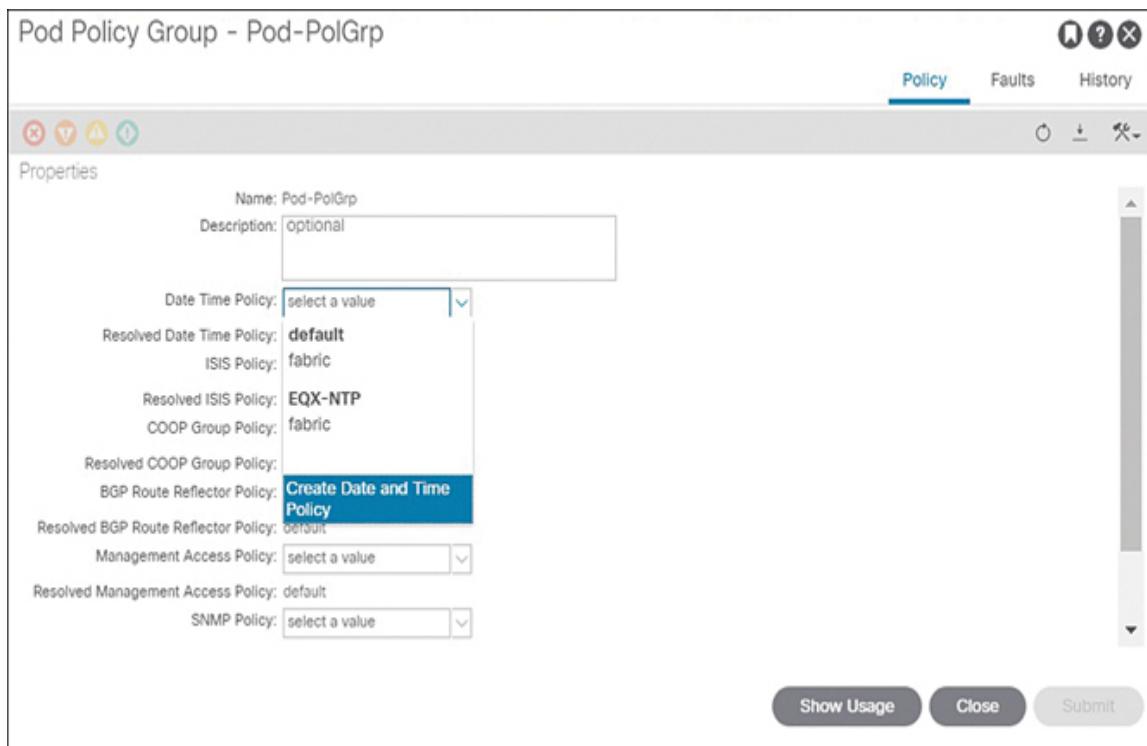


Figure 3-45 Creating a New Date and Time Policy in the Pod Policy Group View

Enter a name for the new policy in the Create Date and Time Policy window and set the policy Administrative State to enabled, as shown in [Figure 3-46](#), and click Next. Note that the Server State parameter allows administrators to configure ACI switches as NTP servers for downstream servers. The Authentication State option determines whether authentication will be required for any downstream clients in cases in which ACI functions as an NTP server.

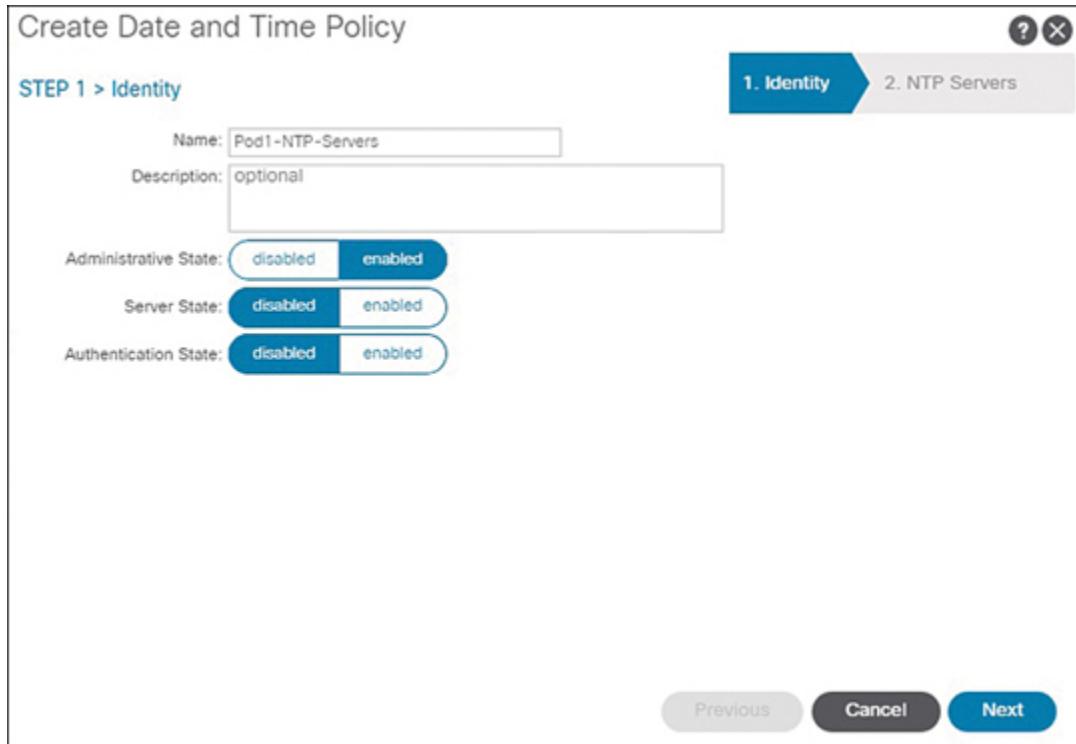


Figure 3-46 Creating a Date and Time Policy

Next, NTP servers need to be defined. Click the + sign on the top-right side of the NTP servers page to create an NTP provider, as shown in [Figure 3-47](#). Enter the IP or DNS address of the NTP server in the Name field and set Minimum Polling Interval, Maximum Polling Interval, Management EPG (in-band or out-of-band) from which communication will be established. Finally, select whether the NTP server being configured should be preferred and then click OK.

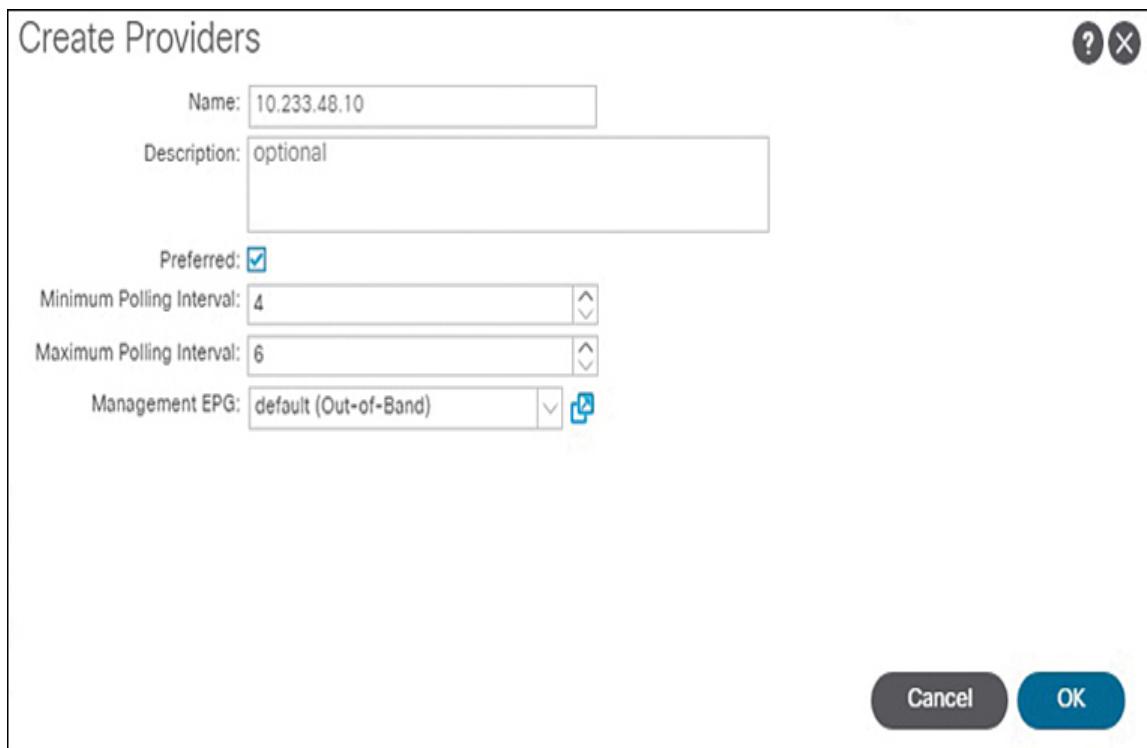


Figure 3-47 Configuring NTP Providers

Once all NTP providers have been configured, as shown in [Figure 3-48](#), select Finish.

Create Date and Time Policy

STEP 2 > NTP Servers

1. Identity 2. NTP Servers

Host Name/IP Address	Preferred	Minimum Polling Interval	Maximum Polling Interval	Management EPG
10.233.48.10	True	4	6	default (Out-of-Band)
10.133.48.10	False	4	6	default (Out-of-Band)

Previous Cancel Finish

The screenshot shows a software interface for creating a Date and Time Policy. It's on 'STEP 2 > NTP Servers'. There are two tabs at the top: '1. Identity' and '2. NTP Servers', with '2. NTP Servers' being active. Below the tabs is a table with columns: Host Name/IP Address, Preferred, Minimum Polling Interval, Maximum Polling Interval, and Management EPG. Two rows are listed: one with IP 10.233.48.10 and another with IP 10.133.48.10. Both rows have 'True' under Preferred, '4' under Minimum Polling Interval, and '6' under Maximum Polling Interval. The Management EPG for both is 'default (Out-of-Band)'. At the bottom are 'Previous', 'Cancel', and 'Finish' buttons.

Figure 3-48 Completing the NTP Provider Configuration

As shown in [Figure 3-49](#), the new date and time policy should appear to be selected in the Date Time Policy drop-down. Click Submit to apply the change.

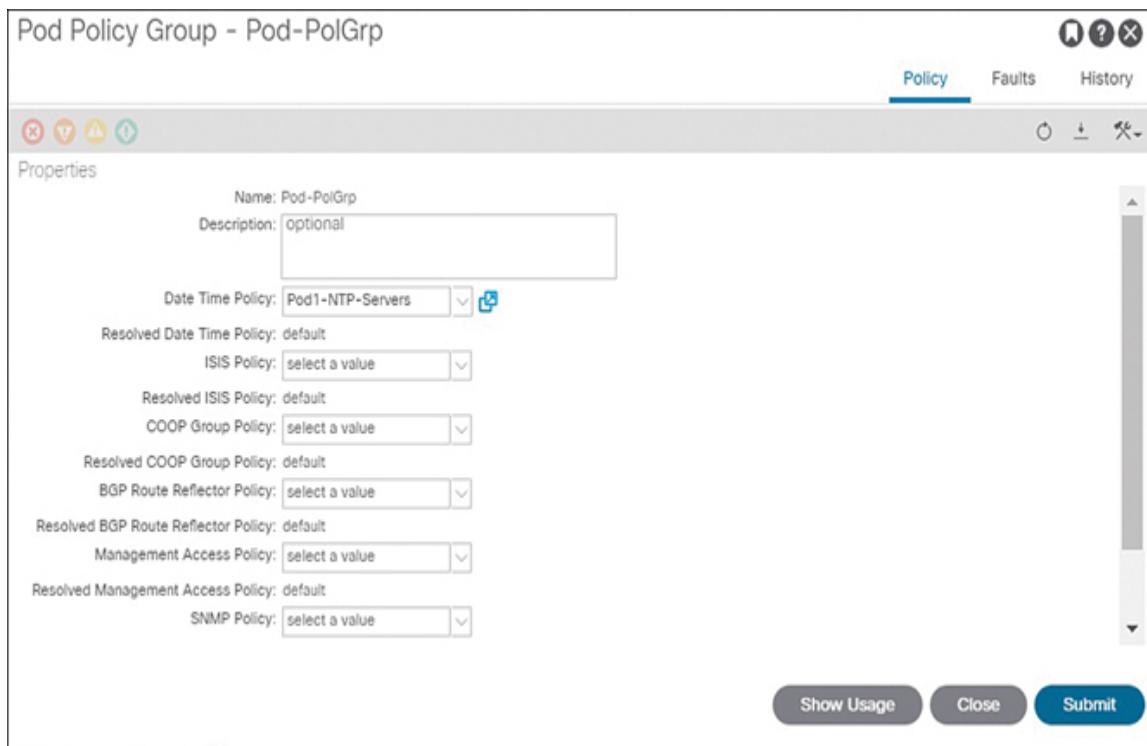


Figure 3-49 Applying Changes to a Pod Policy Group

To verify that the changes have taken effect, log in to the APIC CLI via SSH and run the commands **cat /etc/ntp.conf** and **netstat**, as shown in [Example 3-7](#).

Example 3-7 Verifying NTP Configuration and Synchronization on an APIC

[Click here to view code image](#)

```
apic1# cat /etc/ntp.conf
# Permit time synchronization with our time source, but do
not
# permit the source to query or modify the service on this
system.
tinker panic 501996547
restrict default kod nomodify notrap nopeer noquery
restrict -6 default kod nomodify notrap nopeer noquery
```

```

# Permit all access over the loopback interface. This could
# be tightened as well, but to do so would effect some of
# the administrative functions.
#restrict default ignore
restrict 127.0.0.1
#restrict -6 ::1

keysdir /etc/ntp/
keys /etc/ntp/keys

server 10.233.48.10 prefer minpoll 4 maxpoll 6
server 10.133.48.10 minpoll 4 maxpoll 6

apic1# ntpstat
synchronised to NTP server (10.233.48.10) at stratum 4
    time correct to within 72 ms
    polling server every 16 s

```

Example 3-8 shows how to verify NTP settings on ACI switches. Execution of the commands **show ntp peers** and **show ntp peer-status** on a switch confirms that the APICs have deployed the NTP configuration to the switch and that an NTP server has been selected for synchronization.

Use the command **show ntp statistics peer ipaddr** in conjunction with the IP address of a configured NTP server to verify that the NTP server is consistently sending response packets to the switch.

Example 3-8 Verifying NTP Configuration and Synchronization on an ACI Switch

[Click here to view code image](#)

```
LEAF101# show ntp peers
```

Peer IP Address	Serv/Peer	Prefer
KeyId	Vrf	
10.233.48.10	Server	yes
None management		
10.133.48.10	Server	no
None management		
LEAF101# show ntp peer-status		
Total peers : 3		
* - selected for sync, + - peer mode(active),		
- - peer mode(passive), = - polled in client mode		
remote	local	st
poll reach delay vrf		
LEAF101# show ntp statistics peer ipaddr 10.233.48.10		
remote host:	10.233.48.10	
local interface:	Unresolved	
time last received:	6s	
time until next send:	59s	
reachability change:	89s	
packets sent:	3	
packets received:	3	
bad authentication:	0	
bogus origin:	0	
duplicate:	0	
bad dispersion:	0	

```
bad reference time: 0
candidate order: 0
```

Note that if you know the name of the date and time policy applicable to a pod of interest, you can populate the date and time policy directly by going to Fabric, selecting Fabric Policies, double-clicking Policies, opening Pod, and selecting the desired policy under the Date and Time folder (see [Figure 3-50](#)). If there is any question as to whether the right policy has been selected, you can click the Show Usage button to verify that the policy applies to the nodes of interest.

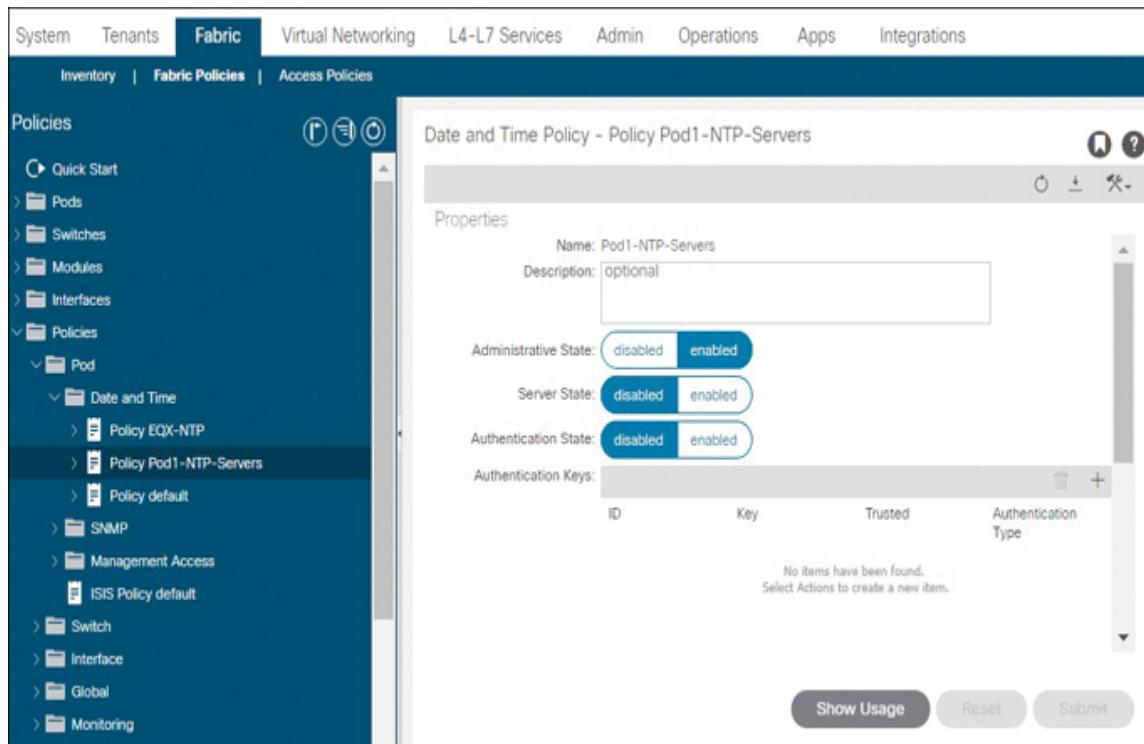


Figure 3-50 Navigating Directly to a Specific Date and Time Policy

If the time for a pod should reflect a specific time zone, the Datetime Format object needs to be modified. You can

modify the Datetime Format object by navigating to System, selecting System Settings, and clicking on Date and Time.

The Display Format field allows you to toggle between Coordinated Universal Time (UTC) and local time. Selecting Local exposes the Time Zone field. Enabling the Offset parameter enables users to view the difference between the local time and the reference time. [Figure 3-51](#) shows the Datetime Format object.

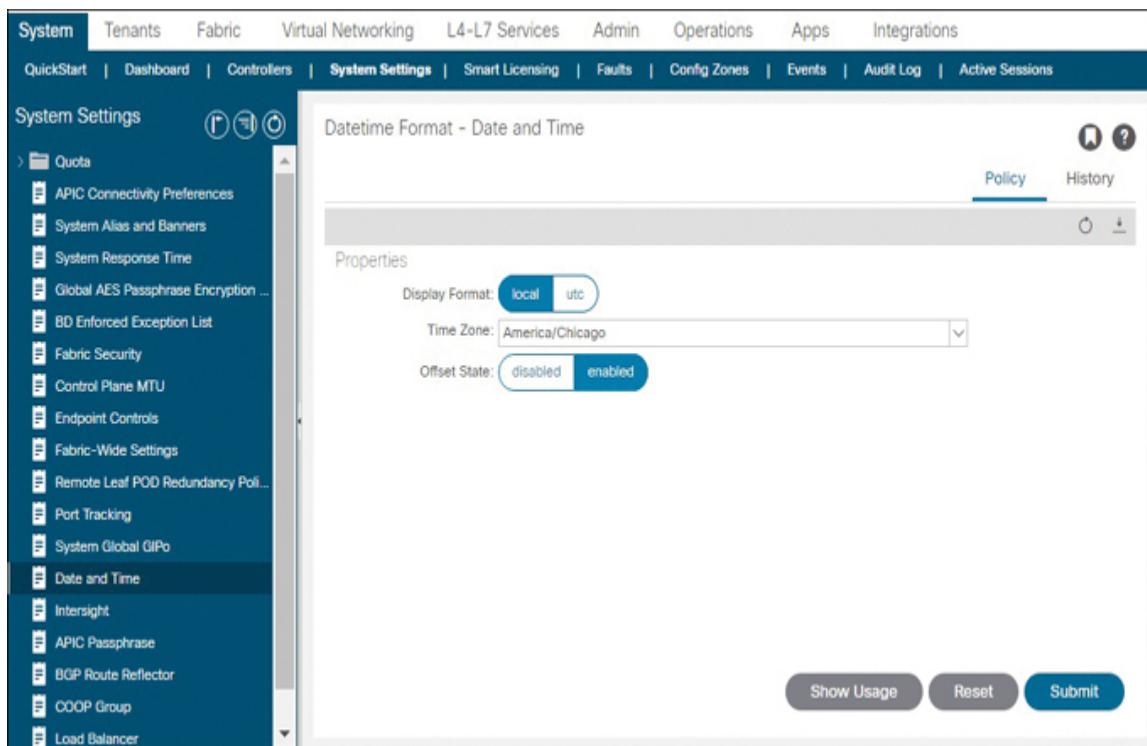


Figure 3-51 Selecting a Time Zone via the Datetime Format Object

Note

NTP is considered a critical service for ACI fabrics. Atomic counters, a capability that measures traffic between leaf switches, requires active NTP synchronization across ACI fabrics. Without NTP

synchronization, ACI is unable to accurately report on packet loss within the fabric.

Configuring DNS Servers for Lookups

Even though DNS is not explicitly within the scope of the DCACI 300-620 exam, DNS is considered a critical service. Various forms of integrations that are within the scope of the exam, such as VMM integration, sometimes rely on DNS. Therefore, this section provides basic coverage of ACI configurations for DNS lookups.

As a multitenancy platform, ACI needs a mechanism for each tenant to be able to conduct lookups against different DNS servers. ACI enables such a capability through DNS profiles. Each profile can point to a different set of DNS servers and leverage a different set of domains.

Administrators can associate a different DNS profile or DNS label to each tenant to ensure that DNS lookups for endpoints within the specified tenant take place using DNS settings from the desired DNS profile.

Where multiple DNS profiles are not needed, a global DNS profile called default can be used to reference corporate DNS servers.

To create a DNS profile, navigate to the Fabric tab, select Fabric Policies, drill into the Policies folder, open the Global folder, right-click DNS Profiles, and select Create DNS Profile. [Figure 3-52](#) shows that the DNS profile name, management EPG (in-band or out-of-band management connections of APICs), DNS domains, and DNS providers should be defined as part of the DNS profile creation process.

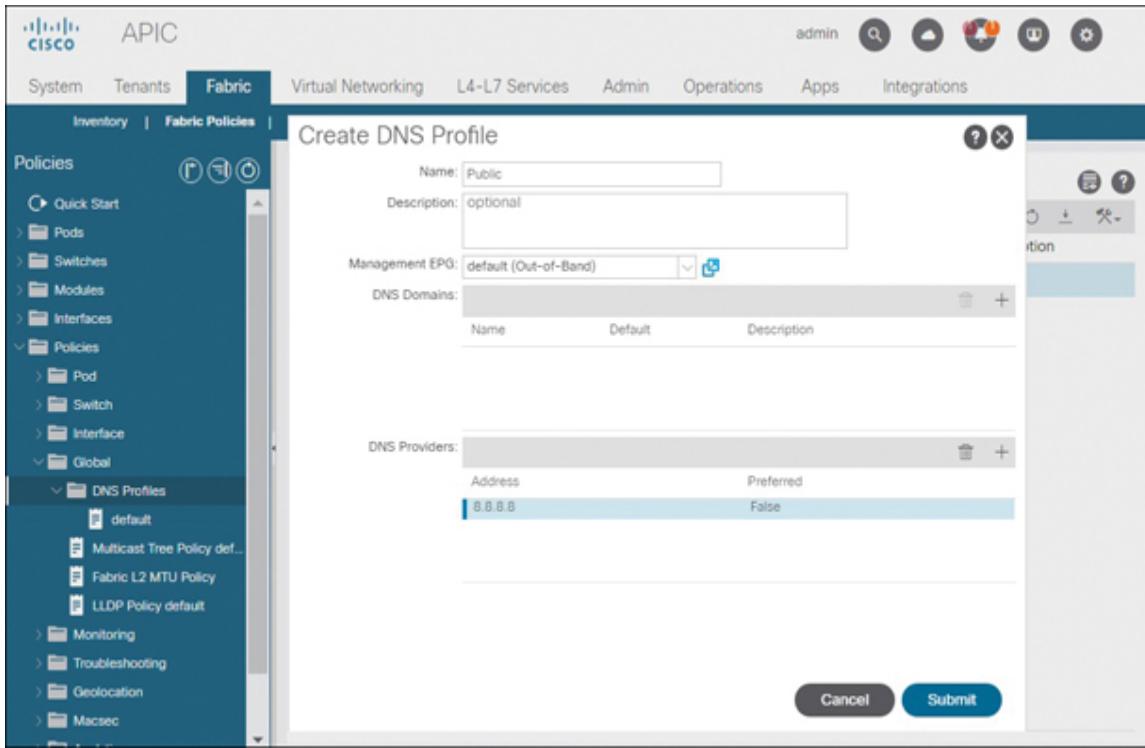


Figure 3-52 Creating a DNS Profile

Once a DNS profile has been created, the DNS label should then be associated with VRF instances within user tenants for ACI to be able to run queries against servers in the DNS profile. [Figure 3-53](#) shows how to assign the DNS label Public to a VRF instance called DCACI within a tenant by navigating to the tenant and selecting Networking, opening VRF instances, selecting the desired VRF instance, clicking on the Policy menu, and entering the DNS profile name in the DNS Labels field.

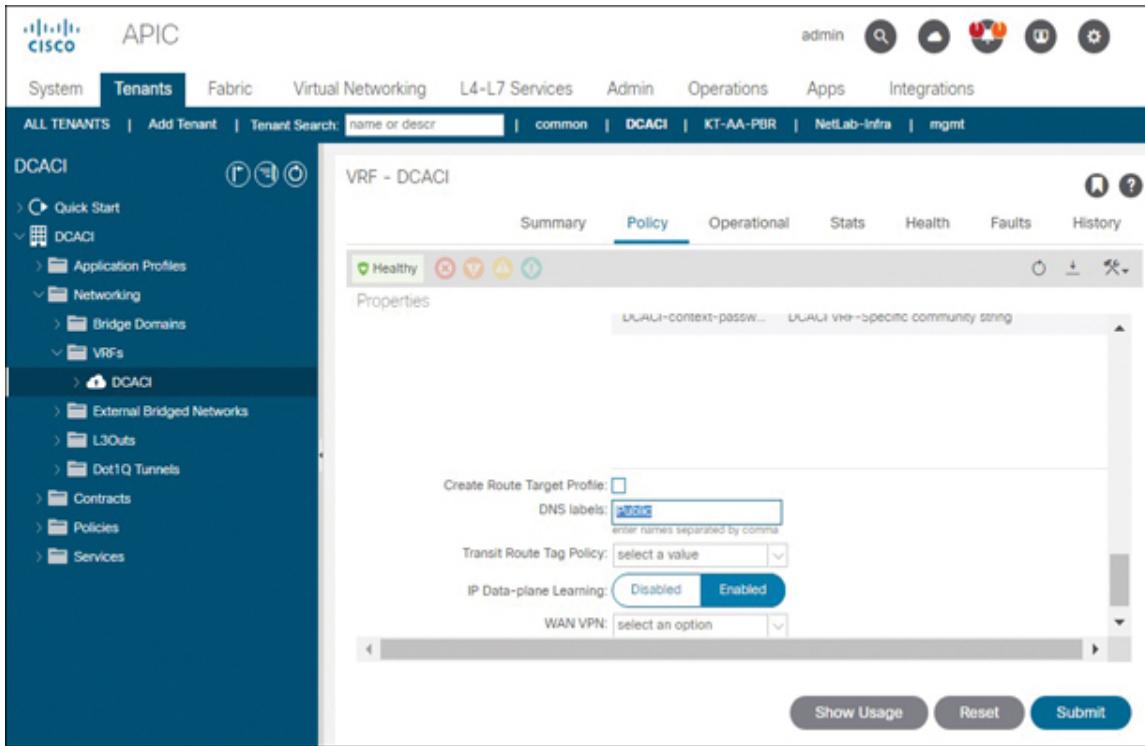


Figure 3-53 Assigning a DNS Label Under a VRF Instance

It is important to differentiate between manually selecting a DNS profile for a user tenant and associating a DNS profile that enables the APICs themselves to conduct global lookups. For the APICs to conduct lookups within the CLI and for use for critical functions, the DNS profile named default needs to be configured, and the label default needs to be associated with the in-band or out-of-band management VRF instances. [Figure 3-54](#) shows the default label being associated with the VRF instance named oob. Association of any DNS label other than default with the inb and oob VRF instances triggers faults in ACI.

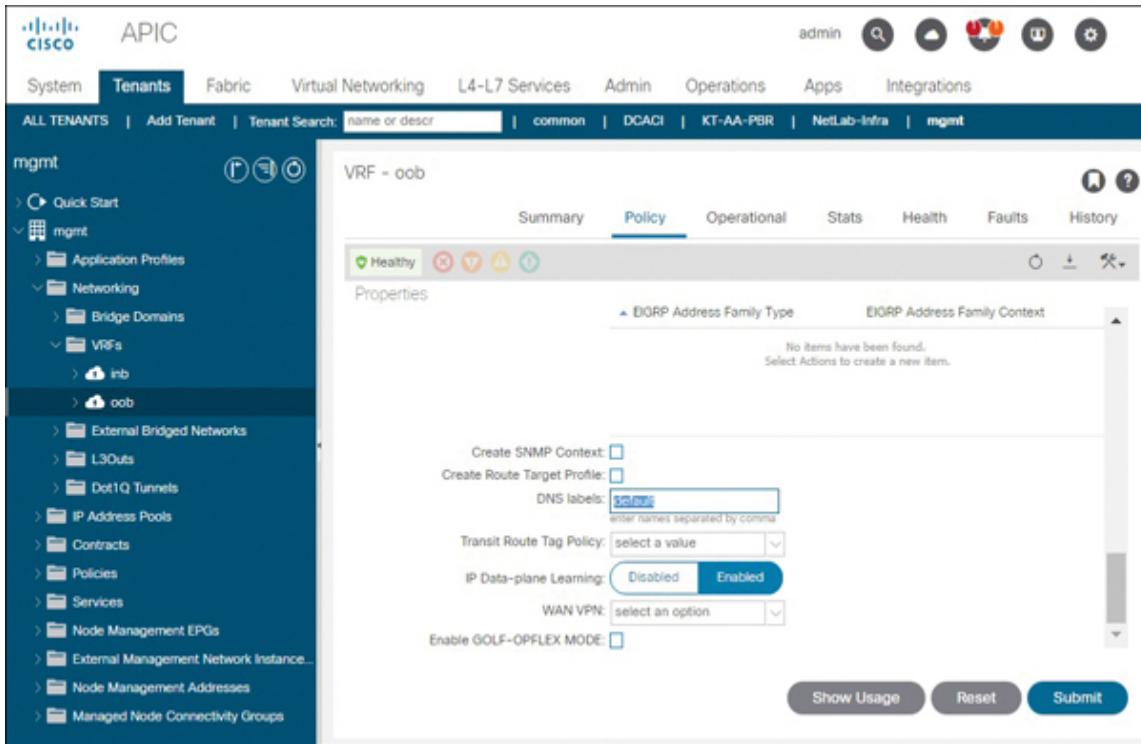


Figure 3-54 Assigning the Default DNS Label to the oob VRF Instance

Following association of the default label to the oob VRF instance, the APICs should be able to execute pings against servers using their fully qualified domain names.

Verifying COOP Group Configurations

Council of Oracle Protocol (COOP) is used to communicate endpoint mapping information (location and identity) to spine switches. A leaf switch forwards endpoint address information to the spine switch Oracle by using ZeroMQ.

COOP running on the spine nodes ensures that every spine switch maintains a consistent copy of endpoint address and location information and additionally maintains the distributed hash table (DHT) repository of endpoint identity-to-location mapping database.

COOP has been enhanced to support two modes: strict and compatible. In strict mode, COOP allows MD5 authenticated ZeroMQ connections only to protect against malicious traffic injection. In compatible mode, COOP accepts both MD5 authenticated and non-authenticated ZMQ connections for message transportation.

While COOP is automatically configured by ACI, it is helpful to be able to see the COOP configuration. To validate COOP settings, navigate to System, select System Settings, and click COOP Group.

[Figure 3-55](#) shows COOP enabled on both spines with the authentication mode Compatible Type within a given fabric. When spines are selected to run COOP, ACI automatically populates the Address field with the loopback 0 address of the spines selected. If enforcement of COOP authentication is required within an environment, you need to update the authentication mode to strict type.

The screenshot shows the Cisco ACI management interface. The top navigation bar includes links for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below this is a secondary navigation bar with links for QuickStart, Dashboard, Controllers, System Settings (which is currently selected), Smart Licensing, Faults, Config Zones, Events, Audit Log, and Active Sessions. On the left, a sidebar titled 'System Settings' lists various configuration options: Quota, APIC Connectivity Preferences, System Alias and Banners, System Response Time, Global AES Passphrase Encryption Settings, BD Enforced Exception List, Fabric Security, Control Plane MTU, Endpoint Controls, Fabric-Wide Settings, Remote Leaf POD Redundancy Policy, Port Tracking, System Global GIPo, Date and Time, Intersight, APIC Passphrase, BGP Route Reflector, COOP Group (which is selected and highlighted in blue), Load Balancer, and Precision Time Protocol. The main content area is titled 'COOP Group Policy - COOP Group'. It displays the 'Policy Property' section with a 'Type' dropdown set to 'Compatible Type'. Below this is a table titled 'Oracle Nodes' showing two entries: Node ID 201 (Node Name SPINE201, Address 10.233.46.33/32) and Node ID 202 (Node Name SPINE202, Address 10.233.46.35/32). At the bottom right are buttons for 'Show Usage', 'Reset', and 'Submit'.

Figure 3-55 Verifying COOP Settings in ACI

Exam Preparation Tasks

As mentioned in the section “How to Use This Book” in the Introduction, you have a couple of choices for exam preparation: [Chapter 17](#), “Final Preparation,” and the exam simulation questions in the Pearson Test Prep Software Online.

Review All Key Topics

Review the most important topics in this chapter, noted with the Key Topic icon in the outer margin of the page. [Table 3-7](#) lists these key topics and the page number on which each is found.



Table 3-7 Key Topics for Chapter 3

Key Description	Page Number
Topic	
Category	
Element	
Requirement	

Key Description Topic Element	Page Number
Paragraph	Describes APIC in-band ports and minimal versus recommended connectivity requirements
Paragraph	Describes APIC OOB ports and connectivity requirements
Paragraph	Contrasts APIC OOB ports with Cisco IMC ports
Table 3-3	Calls out basic configuration parameters that need to be planned for fabric initialization
List	Outlines the steps involved in ACI switch discovery

Key Description Topic Element	Page Number	
List	Describes fabric discovery stages	51
Table 3-4	Describes switch discovery states and what each one means	52
Paragraph	Describes the NIC mode and NIC redundancy settings required for proper fabric discovery	53
Paragraph	Describes the process of assigning OOB management addresses to ACI nodes	63
Paragraph	Explains why it is important to configure entries for APICs in the Static Node Management Addresses folder	64

Key Description	Topic Element	Page Number
Paragraph	Describes how to assign the default contract to the OOB management EPG	64
Paragraph	Outlines what external management network instance profiles are and how they can be used to define external subnets that should be allowed to communicate with ACI from a management perspective	64
List	Recaps the process of assigning an open contract to the out-of-band network	66
Paragraph	Describes how to upload firmware to APICs	67

Key Description	Topic Element	Page Number
Paragraph	Describes how to kick off APIC upgrades	67
Paragraph	Provides additional critical details on executing APIC upgrades	68
Paragraph	Explains how to configure an upgrade group	70
Paragraph	Provides additional critical details on configuring and triggering an upgrade group	70
Paragraph	Explains the use of schedulers in ACI	73

Key Description	Topic Element	Page Number
Paragraph	Describes the process of setting a default firmware version to enforce code upgrades for new switches that are introduced into the fabric	74
Table 3-5	Describes import types	75
Table 3-6	Describes import modes	76
Paragraph	Explains how all switches are by default placed into the default pod	83
Paragraph	Explains pod profiles, pod policy groups, and pod selectors	83

Complete Tables and Lists from Memory

Print a copy of [Appendix C, “Memory Tables”](#) (found on the companion website), or at least the section for this chapter, and complete the tables and lists from memory. [Appendix D, “Memory Tables Answer Key”](#) (also on the companion website), includes completed tables and lists you can use to check your work.

Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

- APIC in-band port
- APIC OOB port
- APIC Cisco IMC
- TEP pool
- infrastructure VLAN
- intra-fabric messaging (IFM)
- physical tunnel endpoint (PTEP)
- fabric tunnel endpoint (FTEP)
- dynamic tunnel endpoint (DTEP)
- scheduler
- pod profile
- pod policy group
- pod selector

Chapter 4

Exploring ACI

This chapter covers the following topics:

ACI Access Methods: This section reviews the methods available for managing and collecting data from ACI.

Understanding the ACI Object Model: This section provides a high-level understanding of the policy hierarchy in ACI.

Integrated Health Monitoring and Enhanced Visibility: This section explores mechanisms ACI uses to communicate problems and system health to users.

This chapter covers the following exam topics:

- 1.2 Describe ACI Object Model
- 1.3 Utilize faults, event record, and audit log

Before diving too deeply into configuration, it is important for an ACI administrator to understand some basics. For example, what are the methods with which one can interact with ACI?

Another important topic is the ACI object model. Everything in ACI is an object in a hierarchical policy model. Each object and its properties can be manipulated programmatically.

Because of the importance of objects in ACI, this chapter touches on the ACI object hierarchy and how administrators can explore this hierarchy. It also provides a high-level understanding of why data pulled from the GUI may be organized slightly differently from the actual ACI object hierarchy.

Finally, this chapter covers how ACI provides feedback to administrators through faults, events, and audit logs.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 4-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Questions.”](#)

Table 4-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
ACI Access Methods	1–3
Understanding the ACI Object Model	4–6

Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

- 1.** Which special login syntax should a user logging in to an ACI CLI use when trying to authenticate to the fabric by using a non-default login domain?
 - a.** No special syntax is needed.
 - b.** *username@ apic-ip-address*
 - c.** *apic# domain\\username*
 - d.** *apic# username@ apic-ip-address*
- 2.** True or false: Using the ACI switch CLI is a suitable means for making configuration changes in an ACI fabric.
 - a.** True
 - b.** False
- 3.** How can an administrator change the acceptable management access protocols and ports in an ACI fabric?
 - a.** Modify the active pod policy group.

- b. Modify the management access policy or policies associated with the active pod policy groups.
 - c. Modify subobjects of the Policies folder under the tenant named mgmt.
 - d. Modify subobjects of the Admin menu.
- 4. Which of the following changes can be made in the Access Policies view?
 - a. Synchronizing switches to an NTP server
 - b. Operationalizing a switch
 - c. Configuring a port channel down to a server
 - d. Integrating ACI into a hypervisor environment
- 5. Which of the following changes can be made in the Fabric Policies view?
 - a. Configuration of interface-level policies
 - b. AAA configurations
 - c. Configuration of policies that impact large numbers of switches in the fabric
 - d. Integration of L4-L7 services into a tenant
- 6. Which of the following tools can be used to query the ACI object hierarchy? (Choose all that apply.)
 - a. MOQuery
 - b. Find
 - c. Grep
 - d. Visore
- 7. Which fault state suggests that some type of user intervention will definitively be required for the underlying fault condition to be resolved?

- a.** Soaking
 - b.** Raised
 - c.** Retaining
 - d.** Raised-Clearing
- 8.** An administrator has resolved the underlying condition for a fault, but the fault has not been deleted from the Faults view. Which of the following steps need to take place for the fault to be deleted? (Choose all that apply.)
- a.** The administrator acknowledges the fault.
 - b.** The fault is deleted after the clearing interval.
 - c.** Faults are immutable and never deleted from the system.
 - d.** The fault is deleted after the retention interval.
- 9.** Which of the following classes governs fabricwide monitoring policies when no corresponding policy exists under the more specific infra or tenant scopes?
- a.** monEPGPol
 - b.** monFabricPol
 - c.** monInfraPol
 - d.** monCommonPol
- 10.** Which of the following objects can be used for periodic reporting of the operational status of a tenant, a pod, or an entire fabric to management teams?
- a.** Faults
 - b.** Health scores
 - c.** Events
 - d.** Audit logs

Foundation Topics

ACI Access Methods

ACI provides three methods for managing an ACI fabric. For programmatic management and data collection, an administrator can use the ACI representational state transfer (REST) application programming interface (API). For more manual configuration, an administrator can use the built-in graphical user interface (GUI) or the command-line interface (CLI). These management access methods are shown in [Figure 4-1](#).

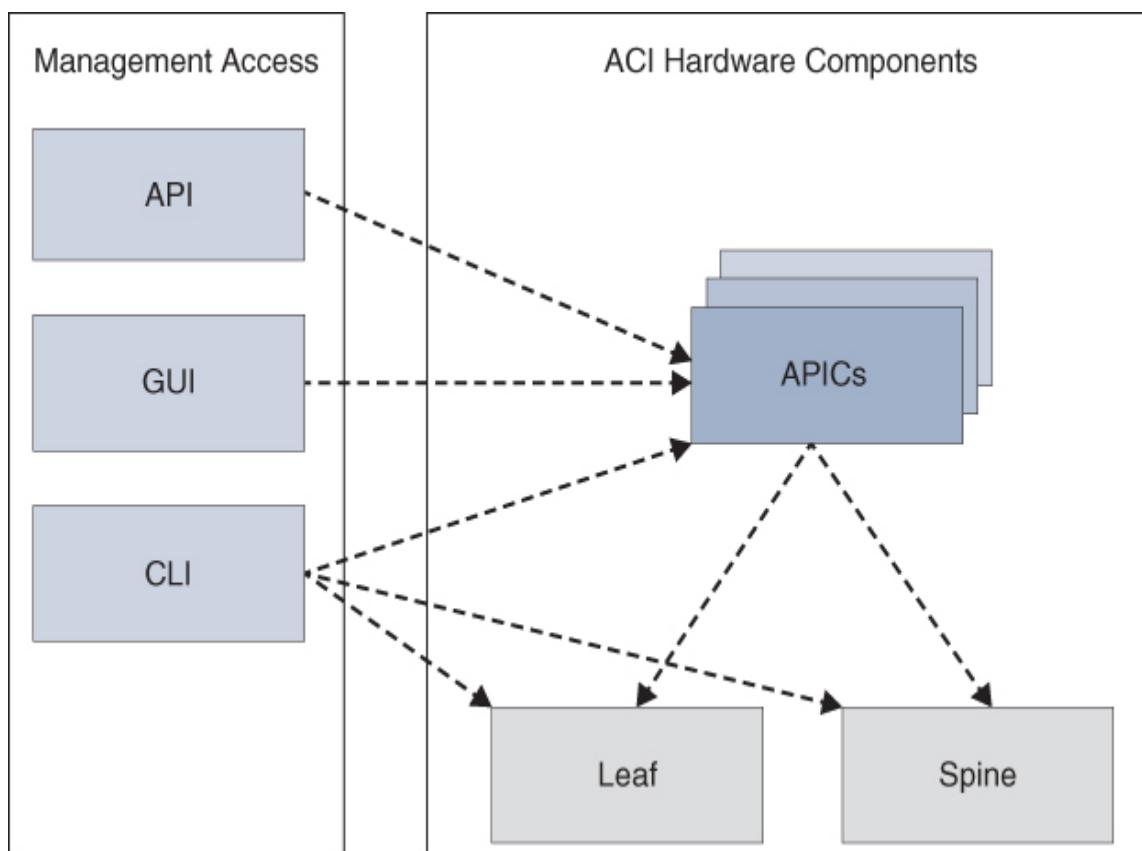


Figure 4-1 Management Access to an ACI Fabric

Administrators configure APICs through one of the three access methods, and all configuration changes are eventually resolved to the API. The GUI and the CLI are basically interfaces into the API.

The process of configuring a fabric involves administrators sending configuration changes to the APICs. The APICs, in turn, deploy necessary configuration changes to leaf and spine switches through an in-band management channel between the APICs and the switches.

GUI

The ACI GUI is an HTML5 application rich with wizards and configuration tools that also enables administrators to verify and monitor an ACI fabric. It is the primary starting point for most newcomers to ACI.

By default, the GUI can only be accessed via HTTPS, but HTTP can be enabled manually, if desired, in lab and low-security environments or in situations in which a backup communication channel to ACI is needed due to HTTPS port changes.

When logging in to the ACI GUI right after fabric initialization, you use the local admin user. However, if you are using a brownfield fabric to familiarize yourself with ACI, chances are that TACACS or some other method for authenticating to the fabric may have been implemented. In this case, the APIC GUI login screen reflects the existence of multiple login domains using a Domain drop-down box like the one shown in [Figure 4-2](#). Sometimes AAA administrators change the default authentication method to reflect the most commonly used authentication domain. In such a case, selecting an explicit login domain from the list may not be necessary. In [Figure 4-2](#), the user explicitly selects the authentication domain named tacacs_domain to

authenticate to a domain other than the default authentication domain.

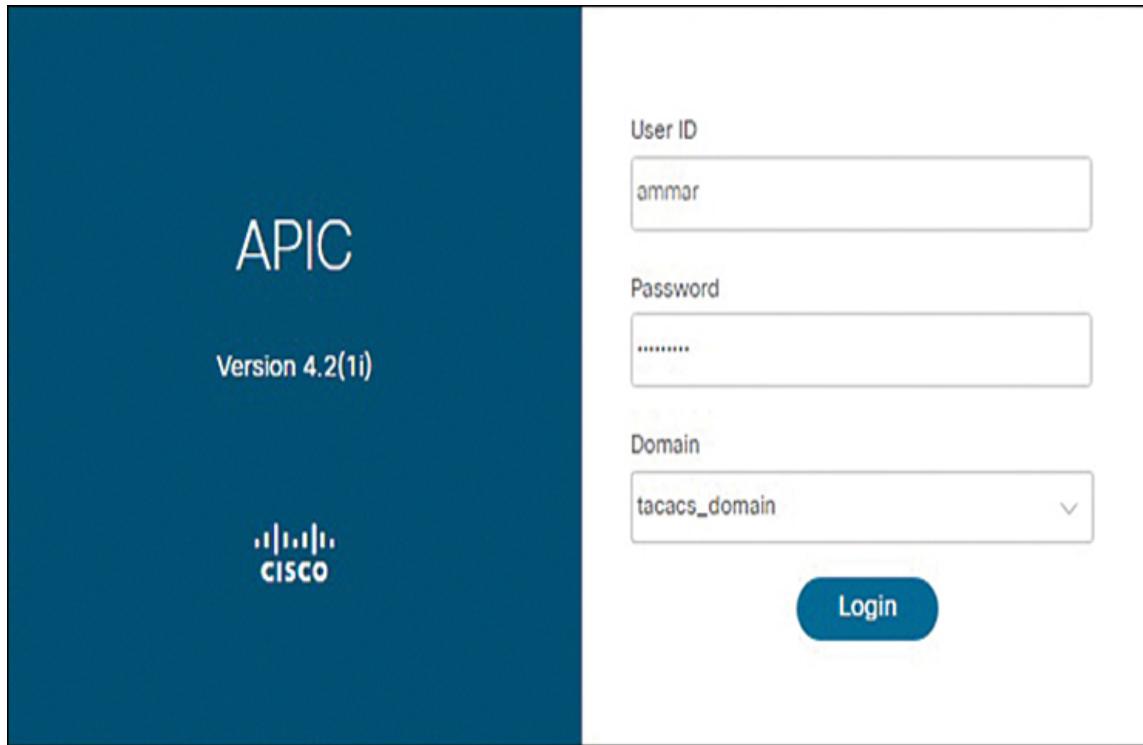


Figure 4-2 Selecting the Right Login Domain in the GUI

Chapter 15, “[Implementing AAA and RBAC](#),” provides a more thorough review of authentication methods and AAA implementation.

CLI

Both APICs and ACI switches offer feature-rich CLIs. The APIC CLI can be used to perform configuration changes in an ACI fabric, but ACI switches cannot be configured directly via the switch CLI. The primary function of the switch CLI is to confirm that intended configurations have been accurately deployed to switches in a fabric and to verify that necessary changes have been programmed into hardware.

For these reasons, the switch CLI is an ideal tool for troubleshooting.

Note

Some configuration changes can be made within an ACI switch CLI. For example, a switch configuration can be restored to its factory defaults by using the script **setup-clean-config.sh** and then reloaded via the **reload** command. Hence, the switch CLI should not be interpreted as a read-only access vector into ACI.

APIC CLI

Administrators can access the APIC CLI directly through the console or by using SSH, Telnet, or the Cisco Integrated Management Console (IMC) Keyboard Video Mouse (KVM) interface with administrative credentials. However, Telnet is disabled by default.

Just as when logging in to an ACI GUI, the local admin user can be used to log in to the APIC CLI after initializing a fabric or if local authentication remains the default authentication avenue configured. However, if AAA settings have been modified and you need to use a non-default domain for authentication, CLI access requires use of a special login format. [Example 4-1](#) shows how the CLI-based username format **apic#domain\\username** can be used to explicitly authenticate by using the desired domain.

Example 4-1 Specifying an Explicit Login Domain for APIC CLI Access

[Click here to view code image](#)

```

login as: apic#tacacs_domain\\ammar
Pre-authentication banner message from server:
| Application Policy Infrastructure Controller
End of banner message from server
apic#tacacs-domain\\dcaci@10.100.5.21's password: <Enter
Password >
apic1#

```

Like other Cisco operating systems, the APIC CLI has multiple configuration modes, as highlighted in [Table 4-2](#).

Table 4-2 Configuration Modes for APIC CLI

Mode	How to Access	Prom pt	Exit Method
Exec	Log in to APIC.	apic1#	exit closes session
Global configuration	From EXEC mode, enter configure .	apic1(config)#	exit moves back to EXEC mode
Configuration submode	From global configuration mode, enter an acceptable command (such as dns).	apic1(config-dns)#	exit moves to the parent end moves to EXEC

mode

When logging in to the APIC CLI, the APIC drops administrators into EXEC mode. The command **configure** or **configure terminal** places administrators into global configuration mode. Execution of certain commands may move the user into a configuration submode.

Example 4-2 shows these concepts. The command **dns** places the user into the DNS configuration submode. The APIC CLI supports the use of question marks to find acceptable commands and arguments for a command. To see a list of commands that begin with a particular character sequence, use a question mark without spaces. Like the majority of other Cisco CLIs, both the APIC and switch CLIs support tab completion. To remove a line from the configuration, use the **no** form of the command in the APIC CLI. The command **end** drops a user into EXEC mode, and **exit** drops the user down one configuration mode or submode.

Example 4-2 Basic Navigation in the APIC CLI

[Click here to view code image](#)

```
apic1# configure
apic1(config)# dns
apic1(config-dns)# show ?
    aaa                  Show AAA information
    access-list          Show Access-list Information
    (...output truncated for brevity...)
apic1(config-dns)# e?
    end                Exit to the exec mode
    exit               Exit from current mode
    export-config      Export Configuration
```

```
apic1(config-dns)# show dns <TAB>
dns-address dns-domain
apic1(config-dns)# show dns-address
Address Preferred
-----
10.100.1.72 no
10.100.1.71 yes
apic1(config-dns)# no address 10.100.1.72
apic1(config-dns)# end
apic1# configure t
apic1(config)# dns
apic1(config-dns)# exit
apic1(config)#
```

Just like NX-OS, the APIC CLI allows users to see a context-based view of the current configuration. [Example 4-3](#) shows a user limiting running configuration output to commands pertinent to DNS.

Example 4-3 Viewing a Specific Portion of the Running Configuration

[Click here to view code image](#)

```
apic1# show running-config dns
# Command: show running-config dns
# Time: Mon Oct 28 14:36:08 2019
dns
  address 10.100.1.71 preferred
  domain aci.networksreimagined.com
  use-vrf oob-default
exit
```

Note

Unlike with NX-OS, IOS, and IOS XE, configuration changes in ACI are automatically saved without user intervention.

Both the APIC and ACI operating systems are highly customized distributions of Linux and therefore support certain Linux functionalities and commands, such as **grep**. In addition, both have a Bash shell that can be used to script and automate tasks. [Example 4-4](#) shows that the Bash shell can be accessed via the **bash** command. You can use the syntax **bash -c 'command'** to execute a Bash-based command while outside the Bash shell.

Example 4-4 Basic Interactions with the Bash Shell

[Click here to view code image](#)

```
apic1# bash
admin@apic1:~>
Display all 1898 possibilities? (y or n)
:
!                               mkmanifest
                               mknbi-dos
 (...output truncated for brevity...)
admin@apic1:~> exit
apic1# bash -c 'uname -ro'
4.14.119atom-3 GNU/Linux
```

Every so often, an ACI user may try to use SSH to access a switch CLI just to find that a management port cable has failed. In such instances, the APIC CLI is your friend. Log in to the APIC CLI and use the **ssh** command followed by the name of the switch in question to establish an SSH session with the switch. This works regardless of the status of the out-of-band management cables to the switches because

these SSH sessions flow over the VICs between the APICs and the switches.

Switch CLI

Unlike the APIC CLI, ACI switches do not support the **show running-config** command. However, a wealth of **show** and **debug** commands is available in the ACI switch CLI.

Unlike the APIC CLI, the ACI switch CLI does not support use of question marks. Instead, you can press the keyboard Tab key twice to leverage tab completion.

Note

ACI switches have several additional CLI shells, such as Bash, that are beyond the scope of the DCACI 300-620 exam. These shells are sometimes beneficial in troubleshooting ACI. They include the following:

- **Virtual Shell (VSH):** The output provided in this CLI mode can sometimes be inaccurate, and this mode has been deprecated.
- **vsh_lc:** This is a line card shell and can be used to check line card processes and hardware forwarding tables.
- **Broadcom shell:** This shell is used to view information on a Broadcom ASIC.

API

Administrators can establish REST API connections with an APIC via HTTP or HTTPS to rapidly configure ACI fabrics via Extensible Markup Language (XML) and JavaScript Object

Notation (JSON) documents. The API can also be used to pull data from an ACI system.

A Python software development kit (SDK) and Python adapter are available for those who want to configure ACI via Python. Ansible can also be used to establish REST API connections and configure ACI via playbooks.

Programmatic configuration and verification of ACI is beyond the scope of the DCACI 300-620 exam and is therefore not covered in this book.

Management Access Modifications

An administrator can enable or disable supported management protocols or change the ports associated with the enabled management access methods. To do so, the administrator needs to edit the active pod management access policy or policies.

To find out which management access policy is active, navigate to the Fabric menu, select Fabric Policies, double-click Pods, select Policy Groups, and then review the available pod policy groups. [Figure 4-3](#) shows a screenshot from an ACI fabric in which only a single pod policy group has been configured. In this figure you can see that Resolved Management Access Policy is set to default. Therefore, to modify enabled management access methods for the pod, you can either navigate to the object or click on the blue link in front of the Management Access Policy pull-down to directly modify the management access methods.

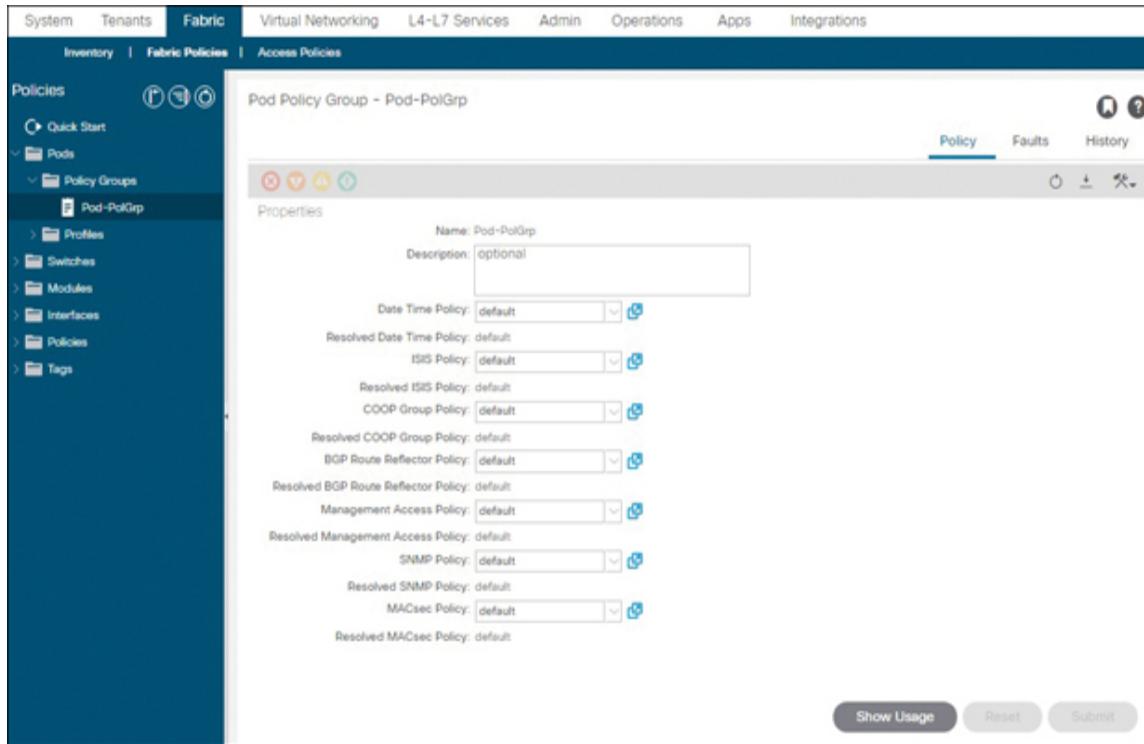


Figure 4-3 Identifying an Active Management Access Policy from the Pod Policy Group Page

If multiple pod policy groups have been configured and each one points to a different management access policy, you can click on the Show Usage button in each pod policy group to review the nodes grouped into that pod policy group. [Figure 4-4](#) shows that nodes 301, 302, 401, and 402 have been associated with a pod policy group named Pod-PolGrp.

Policy Usage Information

? X

i These tables show the nodes where this policy is used and other policies that use this policy. If you modify or delete this policy, it will affect the nodes and policies shown in the tables.

Nodes using this policy			Policies using this policy	
▲ Node Id	Name	Resources	Name	Type
301	LEAF301	n/a	default	Pod Selector
302	LEAF302	n/a		
401	SPINE401	n/a		
402	SPINE402	n/a		

Change Deployment Settings Close

Figure 4-4 Auditing Associated Objects and Nodes via Show Usage

To edit a management access policy, navigate to Fabric Policies under the Fabric menu, double-click on Policies, double-click on Pod, open Management Access, and select the desired policy. [Figure 4-5](#) shows that a user has toggled the Telnet Admin State parameter to Enabled and is ready to click Submit.

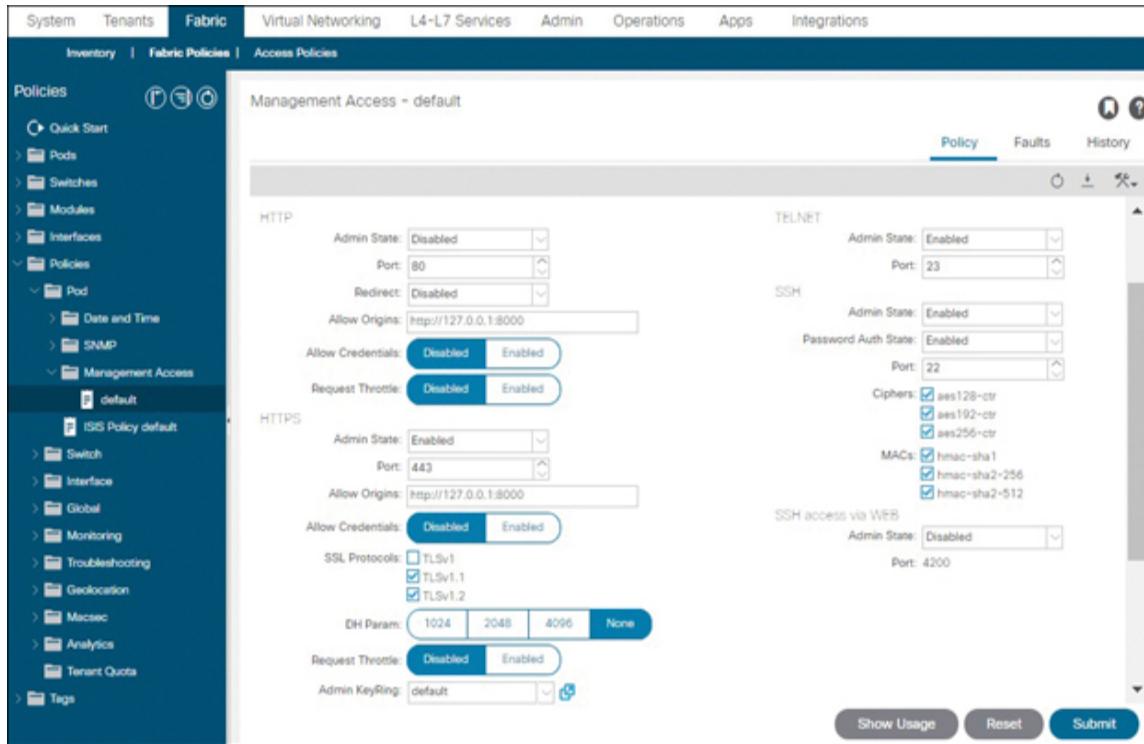


Figure 4-5 Modifying Enabled Management Access Methods in ACI

Notice the SSH Access via WEB option in [Figure 4-5](#). This feature allows you to use the GUI to establish SSH sessions into the APIC over port 4200. This might help in the rare case that port 22 access to APICs has been erroneously blocked by firewalls.

Note

As discussed in [Chapter 3, “Initializing an ACI Fabric,”](#) ACI allows a few default protocols, such as HTTPS and SSH, to access the OOB management interfaces in the fabric. Enabling Telnet as shown in [Figure 4-5](#), however, necessitates that a contract allowing Telnet access as well as its association with an external management network instance profile and the out-of-band or in-band EPG also be defined.

In [Chapter 8](#), “[Implementing Tenant Policies](#),” you will learn more about contract implementation and how to create custom contracts that lock down management access to specified management systems and subnets. For now, the procedure covered in [Chapter 3](#) using the default contract in the common tenant and a 0.0.0.0/0 subnet will suffice in allowing any changes to management access protocols and ports through management access policies.

Understanding the ACI Object Model

An ACI fabric is built using both physical and logical components. These components can be represented in an object hierarchy that is managed by and stored on the APICs and is called the **[Management Information Model \(MIM\)](#)**. The MIM forms a hierarchical tree. The top portion of the tree that represents the bulk of user-configurable policies is called the **[Policy Universe](#)**, as shown in [Figure 4-6](#).

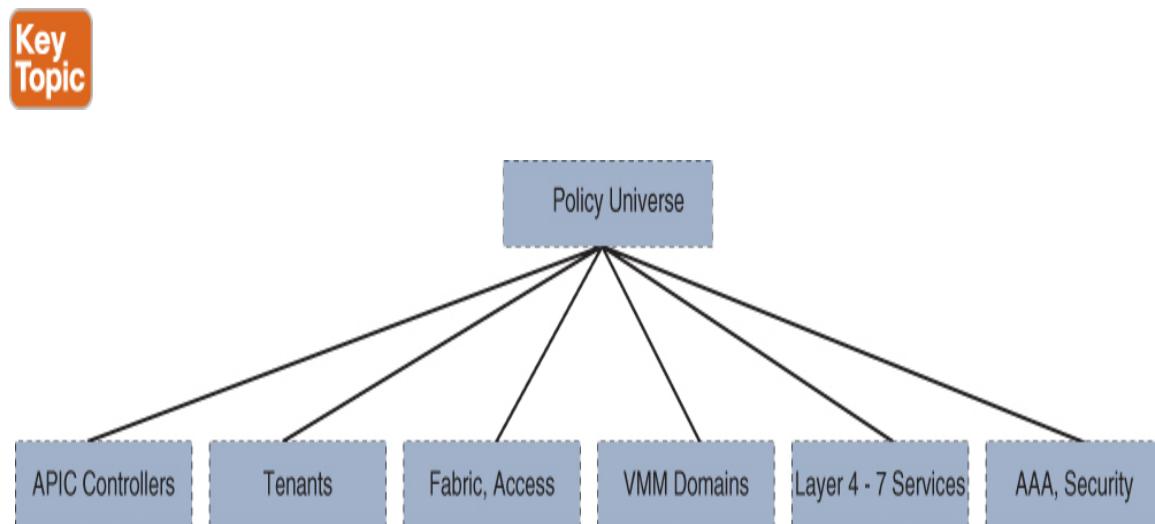


Figure 4-6 *ACI Management Information Model Overview*

Some of the most important branches in the hierarchical tree are the following items:

- **APIC controllers:** An APIC cluster is a clustered set of (usually three) controllers that attaches directly to a leaf switch and provides management, policy programming, application deployment, and health monitoring for an ACI fabric.
- **Tenants:** A tenant is a container for policies that enables an administrator to exercise access control and ensure a level of configuration fault isolation. ACI fabrics come with three predefined tenants. Administrators can create new tenants that are referred to as **user tenants**. Applications as well as Layer 2 and Layer 3 constructs are deployed inside tenants.
- **Fabric policies:** **Fabric policies** govern configurations that apply more holistically at the switch or pod level. Fabric policies also include the operation and configuration of switch fabric ports. Some of the parameters that are configured in the fabric policies branch of the MIM include switch Network Time Protocol (NTP) synchronization, Intermediate System-to-Intermediate System (IS-IS) protocol peering within the fabric, Border Gateway Protocol (BGP) route reflector functionality, and Domain Name System (DNS).
- **Access policies:** **Access policies** primarily govern the configuration and operation of non-fabric (access) ports. Configuration of parameters such as link speed, Cisco Discovery Protocol (CDP), Link Layer Discovery Protocol (LLDP), and Link Aggregation Control Protocol (LACP) for connectivity to downstream servers, appliances, or non-ACI switches, as well as routers all fall into the realm of access policies. Access policies

also include mechanisms to allow or block the flow of tenant traffic on access ports.

- **Virtual networking:** ACI integrations with hypervisor environments referred to as VMM domains as well as integrations with container environments called container domains fall under the umbrella of virtual networking. These types of integrations enable deeper network visibility into virtual environments and policy automation.
- **Layer 4 to Layer 7 Services:** L4-L7 services such as firewalls and load balancers can be integrated to selectively steer traffic to L4-L7 appliances and to enable ACI to dynamically respond when a service comes online or goes offline. L4-L7 services integrations also enable ACI to automatically push configuration changes to devices such as firewalls if they are configured in managed mode.
- **Access, authentication, and accounting (AAA):** AAA policies govern user privileges, roles, and security domains in a Cisco ACI fabric. AAA is a key component of management plane multitenancy in ACI.

Note

The Policy Universe is not the true root in the ACI hierarchy, but it is the portion of the ACI object hierarchy that you are most directly manipulating when you do everyday configuration changes within ACI. Examples of other branches under the real root (topRoot) that reside at the same level as the Policy Universe include topology and compUni.

[Figure 4-7](#) shows several major branches of the ACI object hierarchy that reside directly under topRoot.

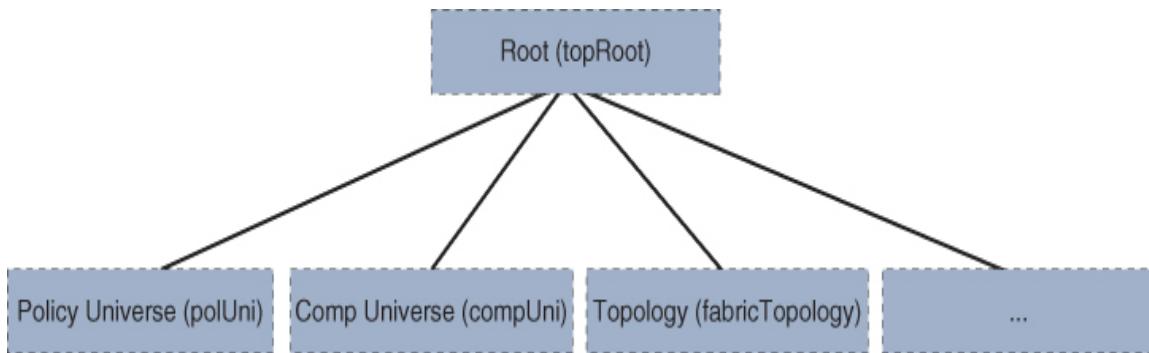


Figure 4-7 Several Branches of the *topRoot* Object

Learning ACI Through the Graphical User Interface

The menus in ACI have a level of correlation to the branches of the management information tree (MIT). This makes it possible to learn the high-level aspects of the ACI object hierarchy just by configuring ACI. [Figure 4-8](#) shows the System submenus, which include the APIC Controllers view.

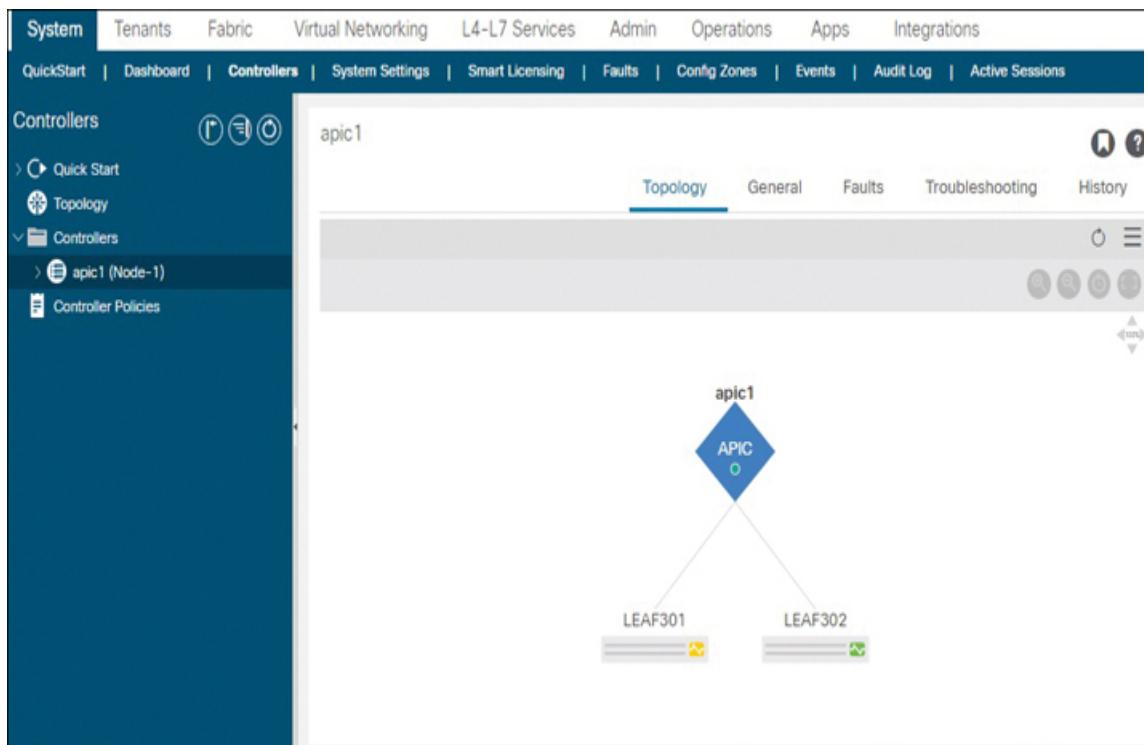


Figure 4-8 APIC Controllers as a System Submenu

The Fabric menu includes the ACI Inventory, Fabric Policies, and Access Policies submenus (see [Figure 4-9](#)).

The screenshot shows the APIC Controller interface with the 'Fabric' tab selected in the top navigation bar. Below the navigation bar, there are three sub-menu items: 'Inventory', 'Fabric Policies', and 'Access Policies'. The 'Inventory' item is currently active, indicated by a blue background. The main content area displays the 'Inventory' sub-menu. On the left, there is a sidebar with icons for 'Quick Start', 'Topology', 'Pod 1', 'Pod Fabric Setup Policy', 'Fabric Membership', 'Disabled Interfaces and Decommissioned', and 'Duplicate IP Usage'. The central panel has a 'Summary' section containing a detailed description of what the Inventory menu does, followed by a 'Steps' section with four items: 'Add Remote Leaf', 'Add Pod', 'Validate the connected switches', and 'Register unregistered switches', each with a corresponding icon. To the right, there is a 'See Also' section with a link to 'Fabric Membership'.

Figure 4-9 Fabric Submenus

[Figure 4-10](#) shows that AAA is a submenu of the Admin menu.

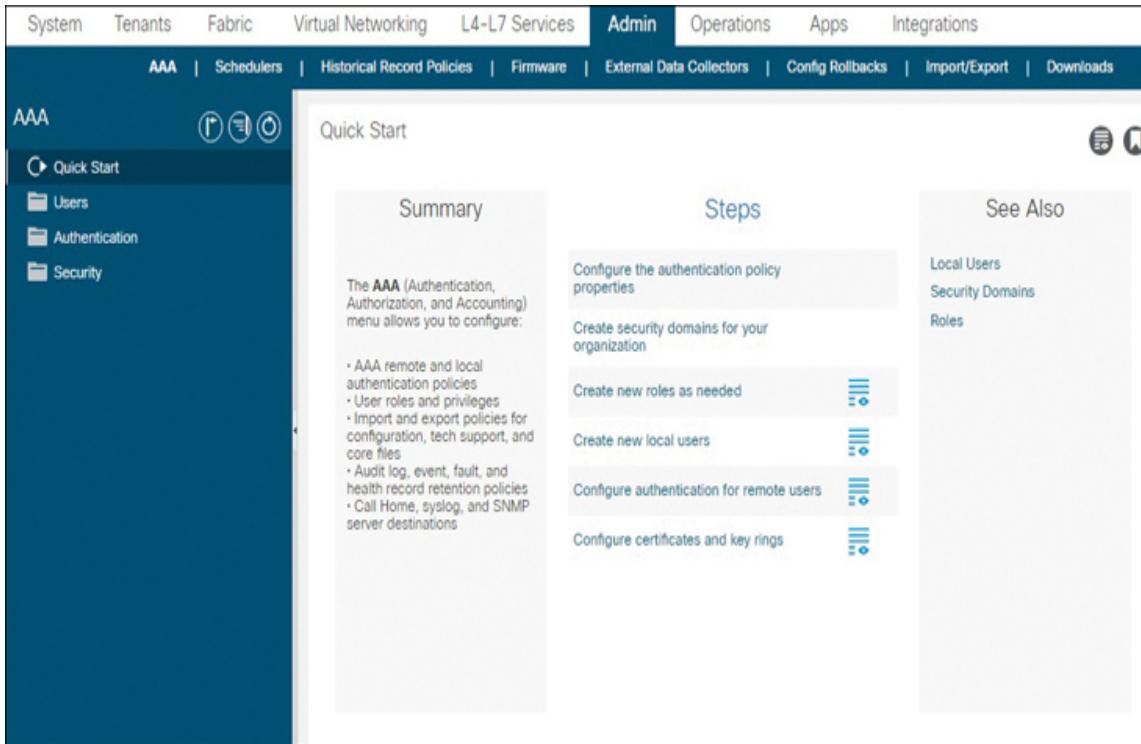


Figure 4-10 Admin Submenus

Exploring the Object Hierarchy by Using Visore

While administrators can use the ACI GUI to learn about the object hierarchy, the menus in the GUI do not translate literally to how objects are placed in the hierarchical tree. In reality, the ACI GUI strives to be user-friendly and easy to navigate and therefore cannot align with the actual object hierarchy.



A tool that can be used to gain a better understanding of the object hierarchy in ACI is **Visore**. Visore can be

accessed by navigating to <https://apic-ip-address/visore.html>, as shown in Figure 4-11.

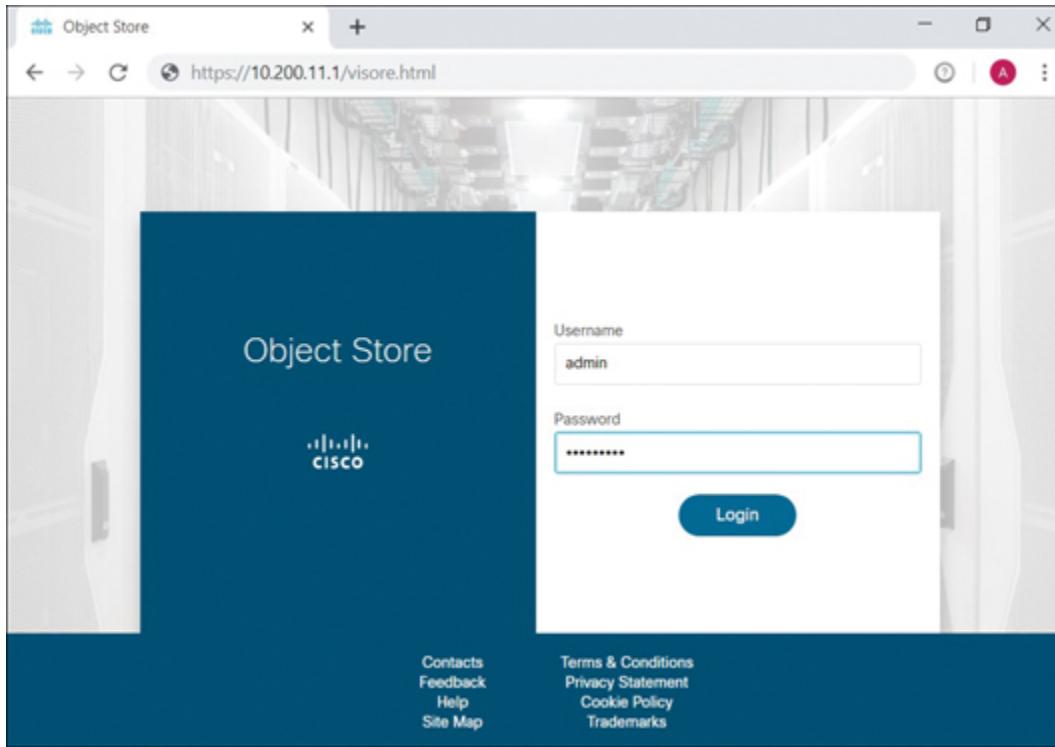


Figure 4-11 Accessing Visore

Once in Visore, you can enter the term **uni** and click the Run Query button to the right of the screen, as shown in Figure 4-12. The object titled polUni appears with the distinguished name uni.

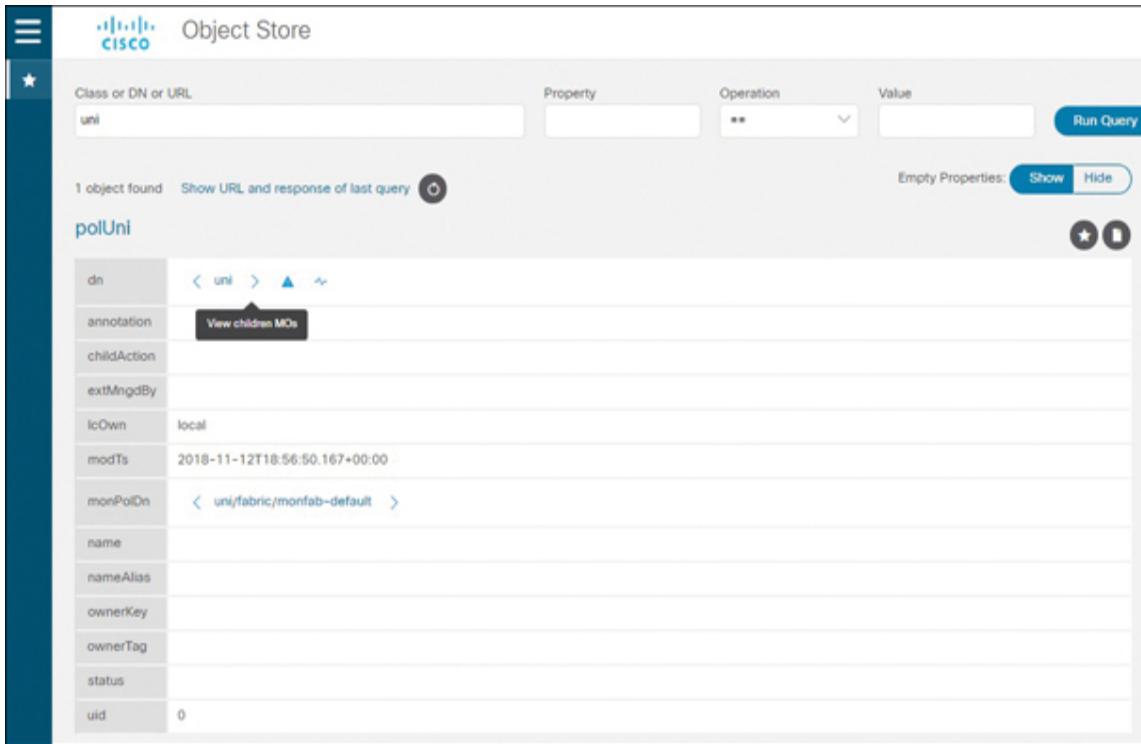


Figure 4-12 Navigating the Policy Universe in Visore



A **distinguished name (DN)** is a unique name that describes an ACI managed object and locates its place in the ACI object hierarchy. Using a DN to search for a specific object is considered an object-level query. A DN query yields zero matches or one match.

To view objects that form direct branches of the policy universe, hover over the arrow pointing to the right and click to examine any direct child objects.

As indicated in [Figure 4-12](#), besides DNs, parameters that can be used in Visore queries are object classes. A *class* refers to one or more objects in the MIM that are of a similar type. For example, the class *fabricNode* refers to all switches and APICs in a fabric. A query of this class,

therefore, could help to quickly glean important information across multiple devices, such as serial numbers. Class-based queries are useful in searching for a specific type of information without knowing any or all of the details. A class-based query yields zero or many results.



An object or a group of objects within the hierarchical tree is called a **managed object (MO)**. MOs are abstractions of fabric resources. An MO can represent a physical object, such as a switch, an interface, or a logical object, such as an application profile, an endpoint group, or a fault.



By navigating the hierarchical tree, you may find that each individual tenant forms its own branch directly under the Policy Universe object. This may not be obvious to engineers who only use the GUI because the GUI places each tenant under a dedicated menu called Tenants.

You can also explore and query the ACI object hierarchy by using an APIC command-line tool called **MOQuery**. You can access help for this tool by using the **moquery -h** command. The **-c** option can be used with **moquery** to query the object model for all objects of a specific class. The **-d** option enables queries based on DN.



Why Understand Object Hierarchy Basics for DCACI?

The purpose of introducing you to the object hierarchy within ACI at this point in the book is more or less to convey how ACI uses a tree structure to enable programmability, to abstract configurations, and to reduce the impact of configuration mistakes.

Note

The separation of tenant configurations from access policy configurations by placing them into separate branches within the hierarchical object tree lowers the scope of impact when making tenant-level configuration changes.

While gaining a detailed understanding of the ACI object hierarchy is important for troubleshooting and automating ACI, it is not necessarily essential for the DCACI 300-620 exam. A rudimentary understanding of the object hierarchy and configuration of ACI via the GUI and the APIC CLI should be sufficient for the DCACI 300-620 exam.

Policy in Context

The word *policy* is used very often in the context of ACI. The term can mean different things in different situations. Because ACI is a policy-driven system, *policy* can refer to almost anything in ACI. In this book, however, if the word *policy* is used independently of any other qualifying words and without further context, it should be interpreted to primarily refer to security policies, forwarding policies, QoS policies, and any other types of policies that specifically center around applications.

Integrated Health Monitoring and Enhanced Visibility

As discussed in [Chapter 1, “The Big Picture: Why ACI?”](#) one of the shortcomings of traditional networks is related to network management, including the difficulty of correlating information across multiple devices in data centers and identifying problems and associated root causes quickly and efficiently.

In traditional data centers, switches typically have no mechanisms for fault management. Ideally, traditional switches are configured to forward syslog messages to one or more syslog servers. Switches and other devices in the data center are then polled by monitoring servers via protocols like SNMP and ICMP. An ideal outcome for most companies is that one or more applications in the network are able to accurately aggregate and correlate available data from all managed endpoints, identify problems in the network, and open a ticket in an IT service management (ITSM) platform to enable troubleshooting. The best-case end result would be for automated mechanisms to resolve problems without human intervention.

The challenge with this approach is that current-day network management tools do not always provide the level of data needed to identify the root causes of issues, and they are also generally not designed with the deep solution-specific data needed to be used as effective proactive monitoring tools.

In addition to enabling standard monitoring capabilities such as syslog and SNMP, ACI introduces integrated health monitoring and enhanced visibility into the data center network, allowing network operators to identify problems

faster when they occur and potentially become more proactive in their approach toward network management.

Some of the features in ACI that enable integrated health monitoring and enhanced visibility include health scores, faults, event logs, and audit logs. The inclusion of these concepts here in this chapter is intentional. Understanding some of the monitoring capabilities of ACI helps to better convey the flexibility of the ACI object model, the methods with which ACI provides feedback to users, and the ease of troubleshooting the system.



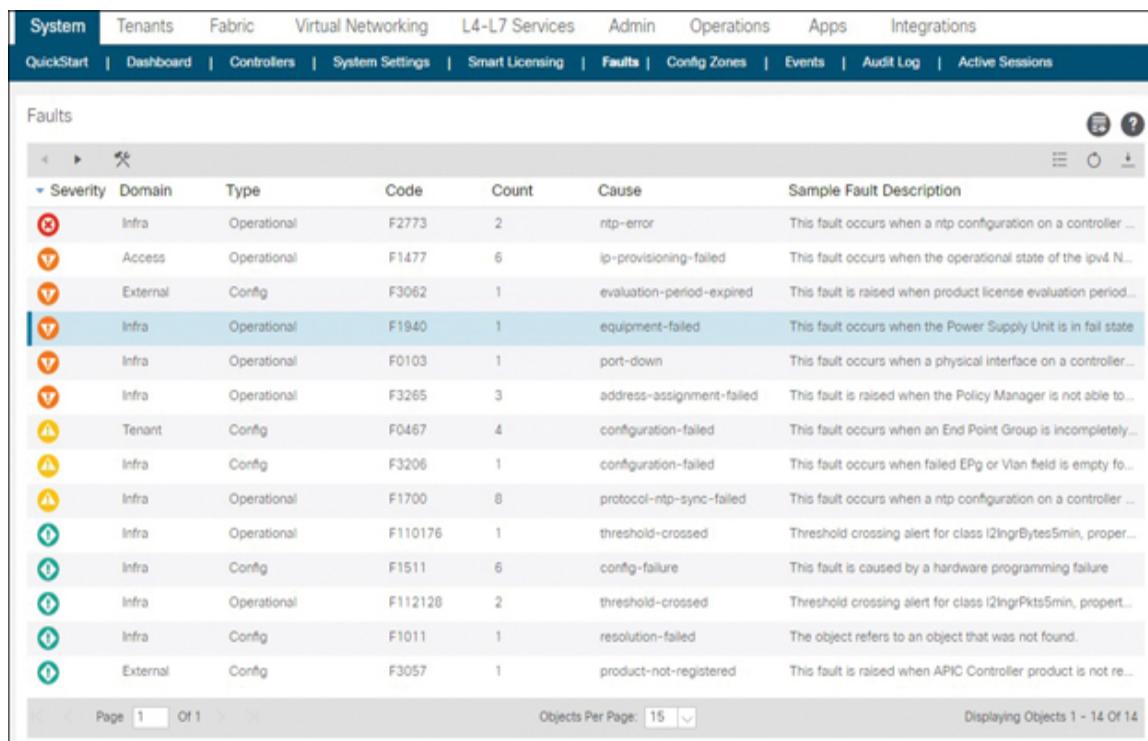
Understanding Faults

A **fault** indicates a potential problem in an ACI fabric or the lack of required connectivity outside the fabric. Each fault has a weight and a severity and is registered into the ACI object hierarchy as a child object to the MO primarily associated with the fault.

Faults in ACI can be created as a result of the following four triggers:

- The failure of a task or finite state machine (FSM) sequence
- Counters crossing defined thresholds, which may, for example, indicate packet loss in the fabric
- Fault rules
- Object resolution failures, which typically result from the deletion of objects referenced by other objects in the fabric

Figure 4-13 shows how active faults in a fabric can be viewed by navigating to **System > Faults**. The leftmost column indicates the faults in order of severity, with red indicating the severity level Critical and green indicating the severity level Warning.



The screenshot displays the 'Faults' section of the Cisco ACI management interface. The top navigation bar includes links for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below this is a secondary navigation bar with links for QuickStart, Dashboard, Controllers, System Settings, Smart Licensing, Faults (which is highlighted in blue), Config Zones, Events, Audit Log, and Active Sessions. The main content area is titled 'Faults' and contains a table with the following columns: Severity, Domain, Type, Code, Count, Cause, and Sample Fault Description. The table lists 14 faults, each with a corresponding severity icon (red for Critical, orange for Major, yellow for Minor, green for Warning). The faults are categorized by domain (Infra, Access, External) and type (Operational, Config). The 'Severity' column uses color-coded icons to represent the fault's severity level. The 'Cause' column provides a brief description of the fault, and the 'Sample Fault Description' column provides a detailed explanation. The bottom of the page includes pagination controls (Page 1 of 1), object per page selection (15), and a note indicating 14 objects displayed out of 14 total.

Severity	Domain	Type	Code	Count	Cause	Sample Fault Description
critical	Infra	Operational	F2773	2	ntp-error	This fault occurs when a ntp configuration on a controller ...
major	Access	Operational	F1477	6	ip-provisioning-failed	This fault occurs when the operational state of the ipv4 N...
major	External	Config	F3062	1	evaluation-period-expired	This fault is raised when product license evaluation period...
warning	Infra	Operational	F1940	1	equipment-failed	This fault occurs when the Power Supply Unit is in fail state
warning	Infra	Operational	F0103	1	port-down	This fault occurs when a physical interface on a controller...
warning	Infra	Operational	F3265	3	address-assignment-failed	This fault is raised when the Policy Manager is not able to...
warning	Tenant	Config	F0467	4	configuration-failed	This fault occurs when an End Point Group is incompletely...
warning	Infra	Config	F3206	1	configuration-failed	This fault occurs when failed EPg or Vlan field is empty fo...
warning	Infra	Operational	F1700	8	protocol-ntp-sync-failed	This fault occurs when a ntp configuration on a controller ...
warning	Infra	Operational	F110176	1	threshold-crossed	Threshold crossing alert for class I2IngrBytes5min, proper...
warning	Infra	Config	F1511	6	config-failure	This fault is caused by a hardware programming failure
warning	Infra	Operational	F112128	2	threshold-crossed	Threshold crossing alert for class I2IngrPkts5min, proper...
warning	Infra	Config	F1011	1	resolution-failed	The object refers to an object that was not found.
warning	External	Config	F3057	1	product-not-registered	This fault is raised when APIC Controller product is not re...

Figure 4-13 Navigating to the Faults View

Table 4-3 shows the available severity levels for faults.



Table 4-3 Fault Severity Levels Users May See in the Faults Page

M>Description
Description
Options

**d
e**

C A service-affecting condition that requires immediate corrective action. For example, this severity could indicate that the managed object is out of service, and its capability must be restored.

a
l

M A service-affecting condition that requires urgent corrective action. For example, this severity could indicate a severe degradation in the capability of the managed object and that its full capability must be restored.

M A non-service-affecting fault condition that requires corrective action to prevent a more serious fault from occurring. For example, this severity could indicate that the detected alarm condition is not currently degrading the capacity of the managed object.

W A potential or impending service-affecting fault that currently has no significant effects in the system. An action should be taken to further diagnose, if necessary, and correct the problem to prevent it from becoming a more serious service-affecting fault.

n
g

I	A basic notification or informational message that is not possibly independently insignificant.
f	
o	
C	A notification that the underlying condition for a fault has been removed from the system and that the fault will be deleted after a defined interval or after being acknowledged by an administrator.
r	
e	
d	

When a fault is generated, it is assigned a fault code, which helps to categorize and identify different types of faults.

Users can use fault codes to research the fault and possible resolutions for the fault. Codes for each fault are shown in the fourth column of [Figure 4-13](#).

Double-clicking on an individual fault within the Faults view allows a user to drill down further into the details of each fault. [Figure 4-14](#) shows the detailed Fault Properties view for the fault highlighted in [Figure 4-13](#). In this case, the fault correctly reflects the fact that a power supply on an APIC has failed or has been removed.

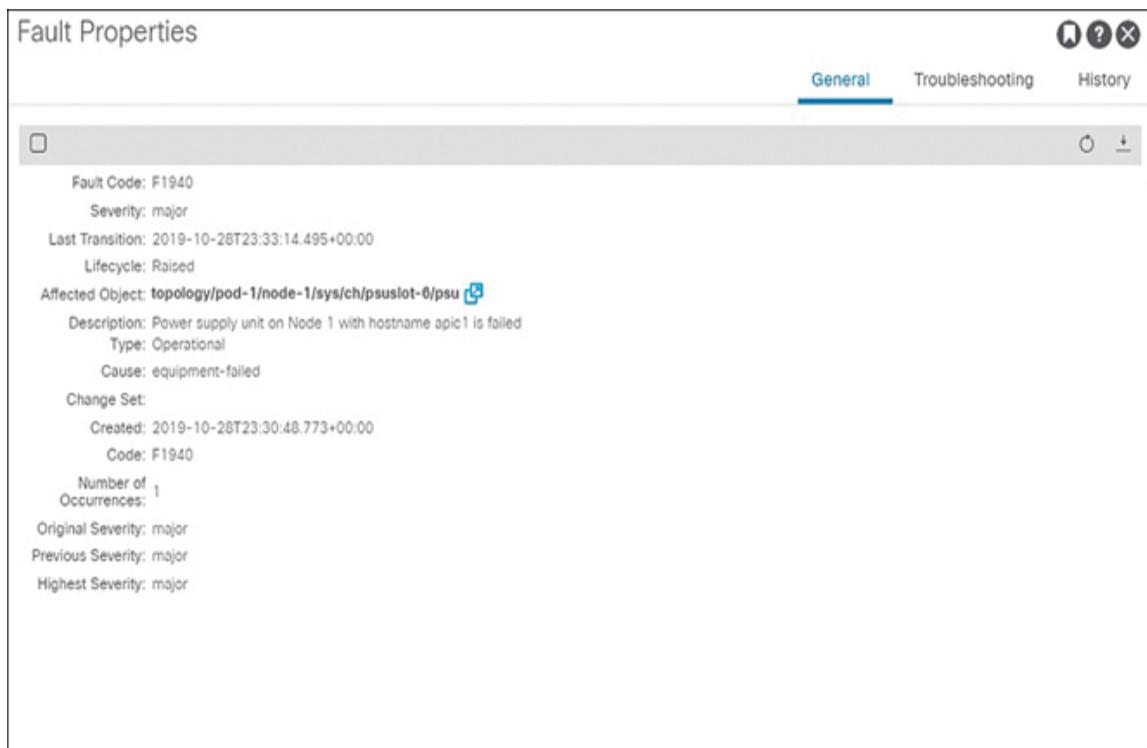


Figure 4-14 *Fault Properties View in the GUI*

Notice that in addition to severities levels, the faults depicted in [Figure 4-13](#) have been categorized into fault types and fault domains. [Table 4-4](#) details the categories of faults that exist in ACI as of Release 4.2.



Table 4-4 Fault Types

Fault Type	Description

Configuration	The system is unable to configure a component as requested by a user.
Environmental	The system has detected a power issue, a thermal issue, a voltage issue, or a loss of CMOS settings.
Management	The system has detected a serious management issue. For example, critical services cannot be started or components within a fabric might have incompatible firmware versions.
Operational	The system has detected an operational issue, such as a log capacity limit having been hit, a link failure, or a component discovery failure.

Domain, in the context of faults, refers to the aspect of the fabric that may be impacted by a fault. For instance, the domain named Security would categorize security-related issues such as lack of connectivity to configured TACACS servers. The domain named Tenant might include faults generated within a specific user tenant. [Figure 4-15](#) shows a view of the GUI dashboard in which faults have been categorized by domain for quick administrator review.

Fault Counts By Domain				
	<input type="checkbox"/> Hide Acked Faults	<input type="checkbox"/> Hide Delegated Faults		
SYSTEM WIDE	2	12	13	13
Access	0	6	0	0
External	0	1	0	1
Framework	0	0	0	0
Infra	2	5	9	12
Management	0	0	0	0
Security	0	0	0	0
Tenant	0	0	4	0

Figure 4-15 *Fault Count by Domain*

Aside from the GUI, administrators are also able to query the APICs from the CLI or via the REST API and view all faults.

The Life of a Fault

Faults follow a lifecycle and transition between the phases shown in [Figure 4-16](#).

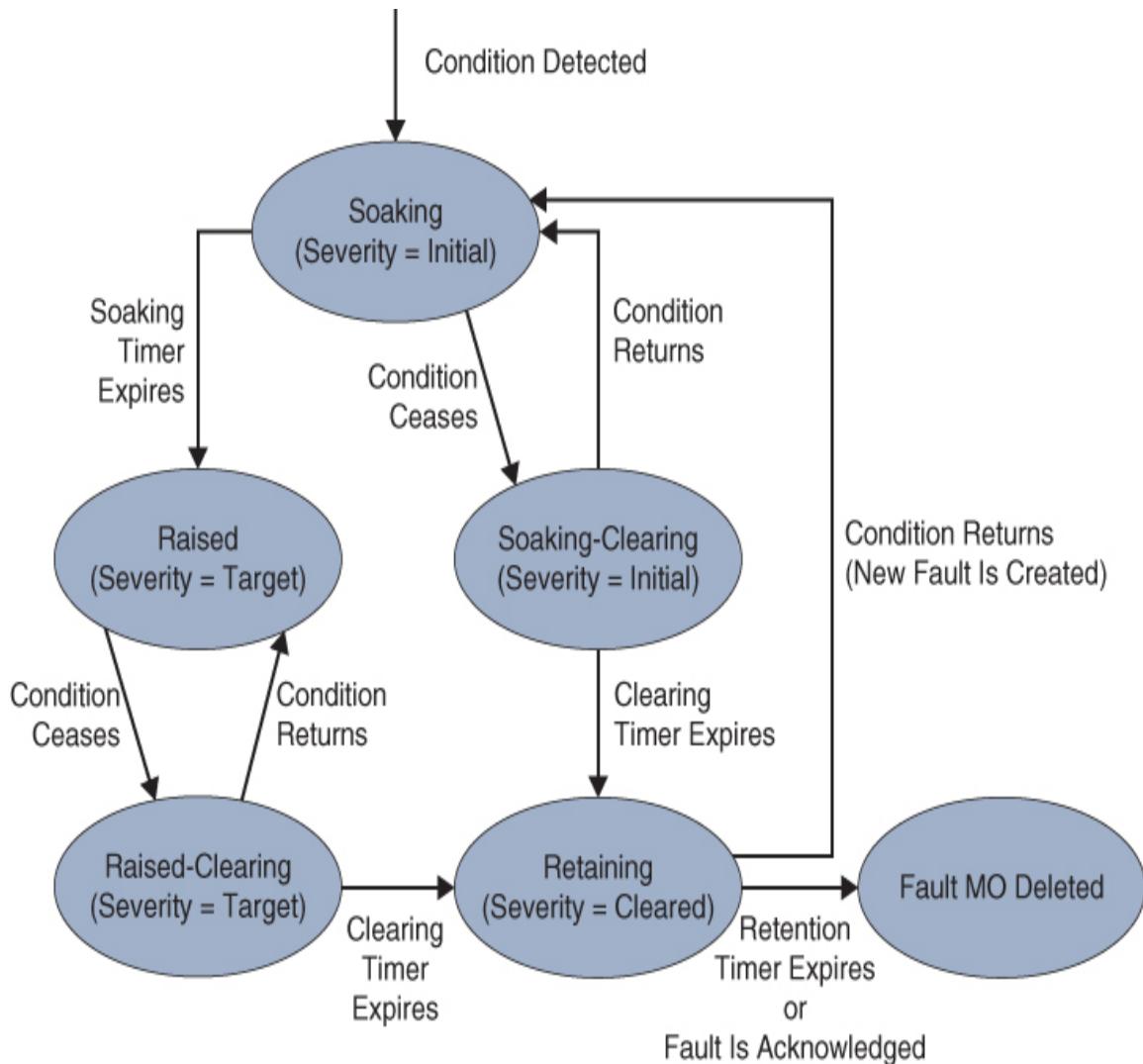


Figure 4-16 Fault Lifecycle

Note

The severity levels initial and target are not included in [Table 4-3](#) earlier in this chapter. The severity level target shown in [Figure 4-16](#) should be understood more as a variable that refers to any severity between Critical and Info. The exact severity for a fault is determined by the default or custom monitoring policies applied to the object experiencing a fault condition.

[Table 4-5](#) provides additional context for [Figure 4-16](#) and details the transitions between fault phases.



Table 4-5 Fault Lifecycle Phases

Description
The initial state of a fault, when a problematic condition is first detected. When a fault is created, the soaking interval begins.

SDescription

t
a
t
e

If a fault condition in the Soaking state is resolved at the end of the soaking interval, the fault enters the Soaking-Clearing state. The fault then stays in this state for the clearing interval. During the clearing interval, if the condition reoccurs, the fault transitions back to Soaking. If not, the fault transitions to the Retaining state.

g
-
C
l
e
a
r
i
n
g

If a fault condition in the Soaking state is not resolved by the end of the soaking interval, it transitions to the Raised state and can have its severity raised. This state suggests the existence of an active problem in the network. Faults in the Raised state remain in this state until the condition is resolved.

SDescription

t
a
t
e

RWhen an administrator addresses a fault condition in the Raised state or when a condition is somehow removed from the system, the fault transitions to the Raised-Clearing state. The clearing interval then begins, and if the condition does not reoccur within this interval, the fault transitions to the Retaining state. If the condition does return within the clearing interval, the fault transitions back to the Raised state.

I
l
e
a
r
i
n
g

SDescription

t
a
t
e

A fault in the Raised-Clearing or Soaking-Clearing state pertaining to a condition that has been absent within the system for the duration of the clearing interval transitions to the Retaining state with the severity level cleared. The retention interval then begins, and the fault remains in the Retaining state for the length of the interval. The fault is deleted either if the condition does not reoccur in this interval or if an administrator acknowledges the fault. If the fault condition reoccurs during the retention interval, a new fault is generated and placed in the Soaking state. The retention interval is generally lengthy, and the goal of this timer is to ensure that administrators are aware of fault conditions that occur in ACI.

Three timers play key roles in the process of transitioning between fault states (see [Table 4-6](#)). These timers can be modified and are defined in fault lifecycle policies.



Table 4-6 Fault Lifecycle Timers

Ti Description

m

er

Cl ea rin g Int er va I	This timer counts the period of time between the system detecting the resolution of a fault condition and the time when the fault severity is set to cleared. This interval refers to the time between the Soaking-Clearing and Retaining fault states. The range for this setting is 0 to 3600 seconds. The default is 120 seconds.
Re te nti on Int er va I	This timer counts the period of time between the system setting the fault severity to cleared and the time when the fault object is deleted. This interval refers to the time between the Retaining fault state and when the fault is deleted. The range for this setting is 0 to 31536000 seconds. The default is 3600 seconds.
So ak in g Int er va I	This timer counts the period of time between ACI creating a fault with the initial severity and the time when it sets the fault to the target severity. This interval refers to the time between the Soaking and Raised fault states. The range for this setting is 0 to 3600 seconds. The default is 120 seconds.

Acknowledging Faults

An administrator can acknowledge a fault when the intent is to delete a fault whose underlying condition has been addressed. To do so, an administrator right-clicks on a given fault and selects Acknowledge Fault, as shown in [Figure 4-17](#).

The screenshot shows the ACI interface under the 'System' tab. In the 'Faults' section, a single fault is listed:

Severity	Acked	Cause	Creation Time	Affected Object	Description	Code	Last Transition	Lifecycle
V		equipm...	2019-10-28T23:30...	topology/pod-1/node-1/sys/ch/psuslot-6/psu	Power supply unit on Node 1 with hostname apic1 is failed	F1940	2019-10-28T23:33...	Raised

A context menu is open over the last row, with 'Acknowledge Fault' highlighted in blue. Other options in the menu include:

- Ignore Fault
- Change Severity
- Save as ...
- Post ...
- Share
- Open In Object Store Browser

Figure 4-17 *Acknowledging a Fault*

ACI then asks the user to confirm the acknowledgment, as shown in [Figure 4-18](#).

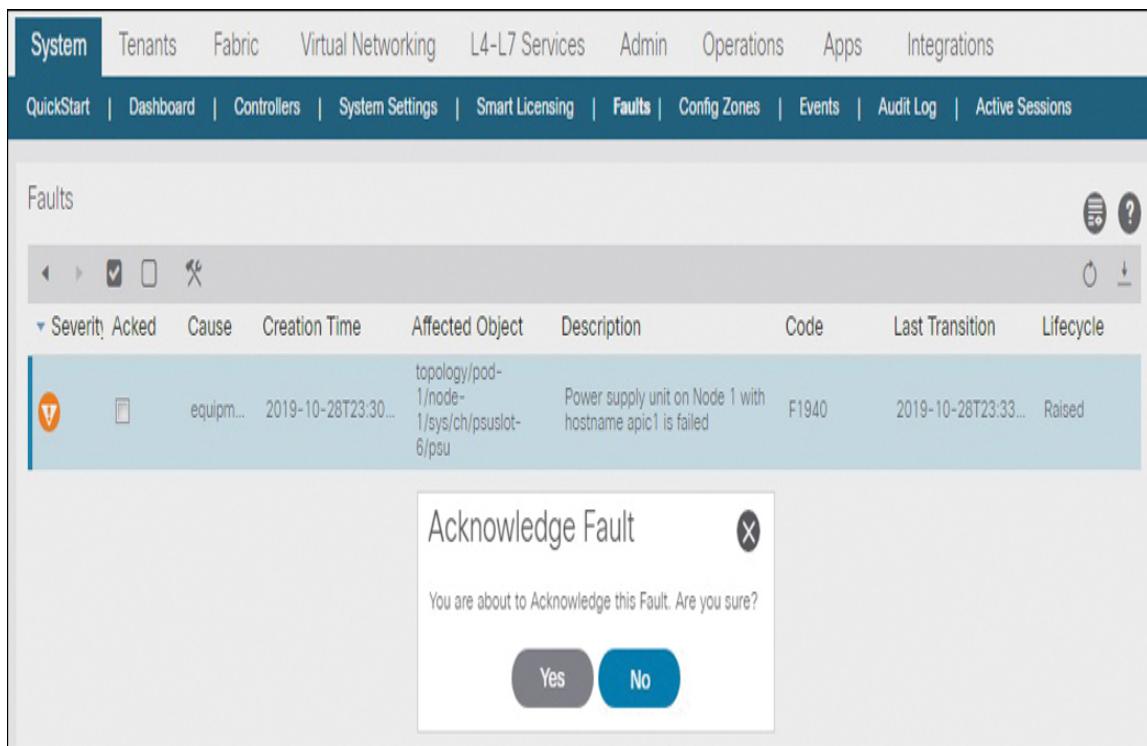


Figure 4-18 *Confirming Fault Acknowledgment*

It is important to understand that the act of acknowledging a fault in ACI may sometimes lead to an instantaneous removal of a fault from the system. However, this is true only when the fault condition has also been removed from the system, and the fault has already transitioned to the Retaining state. In this sense, the acknowledgment is just meant to clear the fault without requiring the Retention Interval to transition to 0.

Acknowledging a fault when the underlying condition remains within the system does not prompt the deletion of the fault. In the case of the fault depicted in Figures 4-17 and 4-18, the underlying condition has not been removed from the system. This is proven by the fact that ACI sees this as an active condition in the Raised state. Although ACI allows the administrator to acknowledge this fault, it does not delete the fault MO until the underlying condition is removed.

Faults in the Object Model

A fault is included in the ACI object hierarchy as an MO of class faultInst or faultDelegate. Fault MOs are usually generated by parent objects. Not all objects can create faults. Objects that *can* generate faults have an attribute called **monPolDn** that points to a monitoring policy object. Figure 4-19 shows an administrator query via Visore of the affected power supply object depicted earlier in this chapter, in Figure 4-14. As shown in Figure 4-19, the monPolDn attribute for this object references the DN uni/fabric/monfab-default. This distinguished name refers to the monitoring policies governing all aspects of fault generation for this object.



Object Store

Class or DN or URL	Property	Operation	Value	Run Query
topology/pod-1/node-1/sys/ch/psuslot-6/psu		==		Run Query
1 object found Show URL and response of last query				
eqptPsu				
dn	< topology/pod-1/node-1/sys/ch/psuslot-6/psu >			
alarmReg	0			
cap	0.000000			
childAction				
descr	PSU2 (ID 7)			
drawnCurr	0.000000			
fanOpSt	unknown			
hwVer	A			
id	2			
mfgTm	not-applicable			
modTs	2019-07-23T15:35:38.153+00:00			
model	UCSC-PSU1-770W			
monPolDn	< uni/fabric/monfab-default >			
operSt	shut			

Figure 4-19 Visore Query of an Affected Object from an Earlier Fault

Using this type of exploration of the policy model, an administrator can begin to gain a deep understanding of how ACI works. Further exploration might lead the administrator to click the link referencing the object `uni/fabric/monfab-default` in Visore. As shown in [Figure 4-20](#), this object is actually of a specific class called `monFabricPol` and has the name `default`.

The screenshot shows the Cisco Visore Object Store interface. At the top, there is a search bar with the URL `topology/pod-1/node-1/sys/ch/psuslot-6/psu`. Below the search bar, it says "1 object found". The object is identified as `monFabricPol`. The details pane shows the following properties:

Property	Value
dn	<code>< uni/fabric/monfab-default ></code>
annotation	
childAction	
descr	
extMngdBy	
lcOwn	local
modTs	2019-04-05T21:30:17.836+00:00
monPolDn	<code>< uni/fabric/monfab-default ></code>
name	default

Figure 4-20 Fabric Monitoring Policy Object in Visore

It turns out that there are four different classes of monitoring policies in ACI that govern aspects of fault and event generation within the fabric. The following section describes them.

Monitoring Policies in ACI

There are four classes of monitoring policies in ACI. You can use the classnames presented in [Table 4-7](#) to query the ACI object hierarchy via Visore and find lists of all configured monitoring policies of the desired class.



Table 4-7 Classes of Monitoring Policies

Mo Description	
monInfraPol	A class of policies that deals with monitoring of infra objects, which includes monitoring of VMM domains, access ports, and external fabric connectivity. Navigate to Fabric > Access Policies > Policies > Monitoring to configure monitoring policies of the monInfraPol class. By default, all infra objects point to the monInfraPol monitoring policy called default. The DN for this default infra monitoring object is uni/infra/moninfra-default.

	<p>mo A class of policies that deals with monitoring of fabric objects, which includes monitoring of fabric uplinks.</p> <p>bri Navigate to Fabric > Fabric Policies > Policies > Monitoring to configure monitoring policies of the monFabricPol class. By default, all fabric objects point to the monFabricPol monitoring policy called default. The DN for this default fabric monitoring policy is uni/fabric/monfab-default.</p>
	<p>mo A policy class that has a global fabricwide scope and deals with monitoring of objects such as the APIC controllers and fabric nodes. The policies configured in this class are also used when there is no corresponding policy under the more specific infra or tenant scopes.</p> <p>bri Navigate to Fabric > Fabric Policies > Policies > Monitoring > Common Policy to modify the common monitoring policy. The DN for the common monitoring policy is uni/fabric/moncommon.</p>
	<p>mo A class of policies that deals with monitoring of tenant objects. Navigate to Tenants, select a tenant, and double-click Policies and then Monitoring to configure monitoring policies of the monEPGPol class. By default, all tenant objects point to the monEPGPol monitoring policy called default. The DN for this default tenant monitoring policy is uni/tn-common/monepg-default. This DN refers to a monitoring policy that resides in a tenant called common. Custom tenant-specific monitoring policies can be created and assigned to tenant objects, if desired.</p>

Note that each of the four classes of monitoring policies outlined in [Table 4-7](#) references a default policy that monitors pertinent objects out of the box. Default monitoring policies can be overridden by specific policies. For example, a user might create a specific monitoring policy within a tenant and associate this new custom policy at the tenant level to override the default policy for the tenant. Customizing the monitoring policy for a specific tenant does not delete the default policy that resides at `uni/tn-common/monepg-default`. Other tenants still reference this default monitoring policy.

One key reason monitoring policies have been introduced in this chapter is to complete the subject of the ACI object model and the use of Visore for developing a better understanding of how ACI works. [Figure 4-21](#), for example, shows how you can right-click almost any object in the GUI and then select Open in Object Store Browser to determine the DN of the object and how other objects may be associated with the object in question. By digging into details like these, you can start to grasp the bigger picture of how policies link together. This figure shows the user trying to find the object representing a fabric port in Visore.

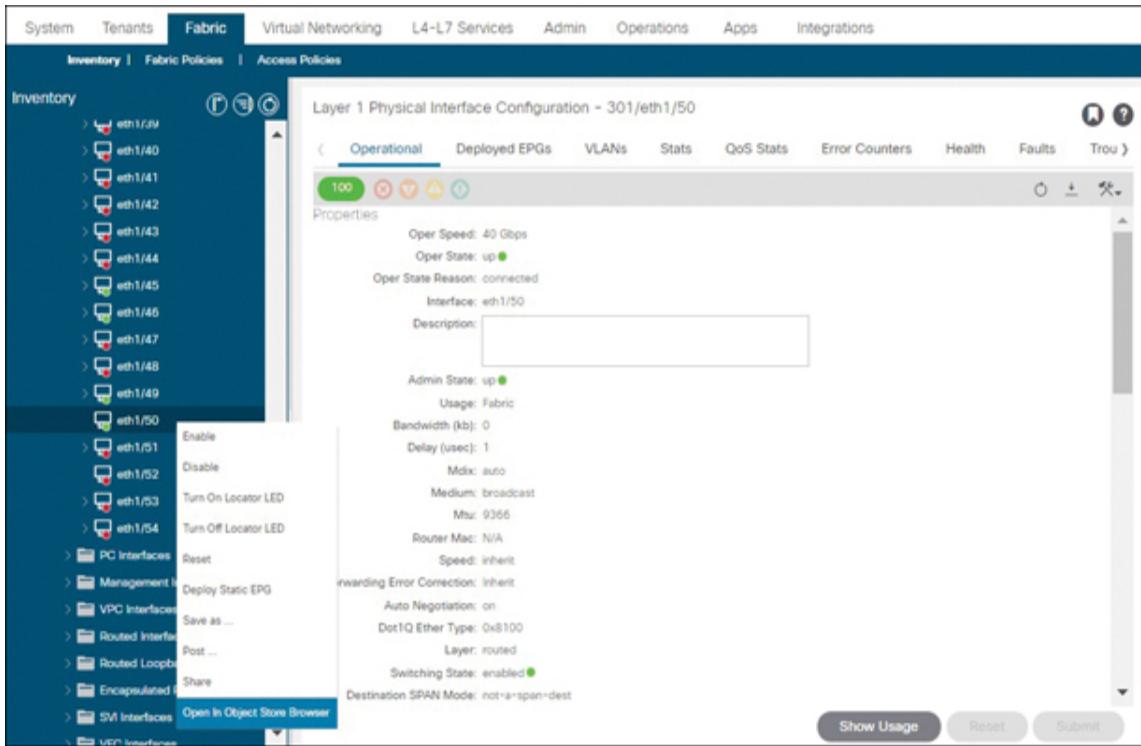


Figure 4-21 Using the GUI to Find an Object in Visore

Figure 4-22 shows that the fabric port in question has the monitoring policy uni/fabric/monfab-default associated with it. This should reinforce the fact that the monitoring class monFabricPol is what determines the monitoring policy for fabric ports.

id	eth1/50
inhBw	unspecified
isReflectiveRelayCfgSupported	Supported
layer	Layer3
lcOwn	local
linkDebounce	100
linkLog	default
mdix	auto
medium	broadcast
modTs	2019-10-29T04:39:01.033+00:00
mode	trunk
monPolDn	uni/fabric/monfab-default < >
mtu	9366

Figure 4-22 Finding the Monitoring Policy (*monPolDn*) Associated with a Port

If the DN being displayed is not enough of a hint, you can navigate to the actual DN and determine the class of the specified DN and correlate that with the various monitoring policies in the fabric.

Note

Some of the concepts in this chapter may seem overwhelming. However, the goal is for you to be able to understand the more theoretical concepts of the object model. By the end of this chapter, you should also be able to identify the various types of monitoring policies and the high-level purposes of each. In addition, you should be able to edit the default monitoring policies and make minor changes. Another important skill you should have developed by the end

of this chapter is to be able to use Visore to figure out which monitoring policy applies to any given object.

Customizing Fault Management Policies

A ***fault lifecycle policy*** specifies the timer intervals that govern fault transitions between states in the lifecycle. Fault lifecycle policies can be specified in the Common policy, within default policies, or in a custom monitoring policy.

To change the timer values configured in the fabricwide Common policy, navigate to Fabric, select Fabric Policies, open up the Policies folder, double-click Monitoring, double-click Common Policy, and select Fault Lifecycle Policy, as shown in [Figure 4-23](#). Then select the new interval values and click Submit.

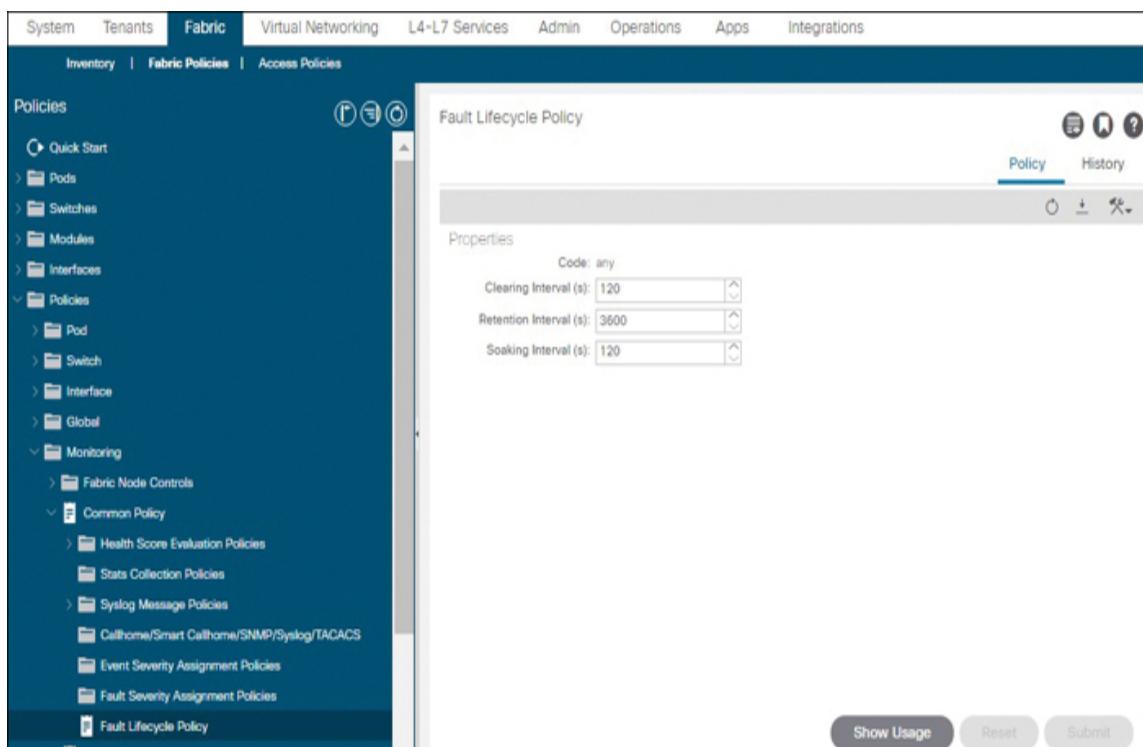


Figure 4-23 Modify Fault Lifecycle Timer Values in the Common Policy

To customize fault lifecycle timer values for default or custom monitoring policies, navigate to the intended monitoring policy, right-click it, and select Create Fault Lifecycle Policy, as shown in [Figure 4-24](#).

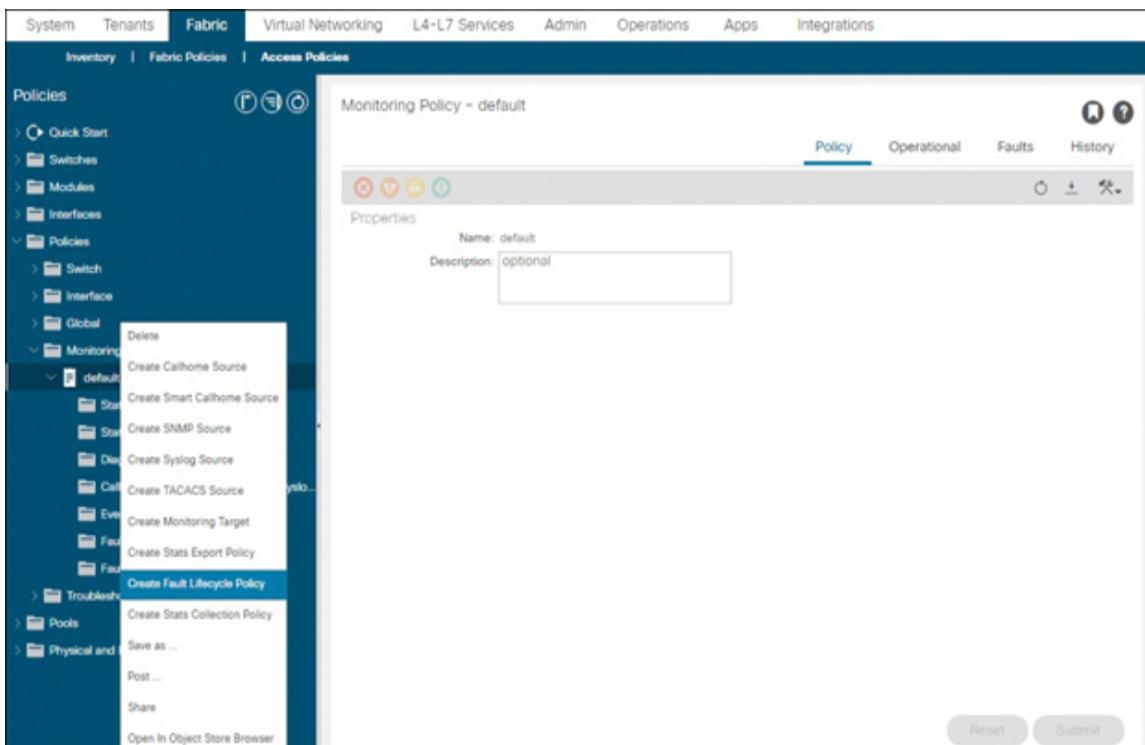


Figure 4-24 Creating a Fault Lifecycle Policy for a Default or Custom Monitoring Policy

Finally, select the desired timer values and click Submit, as shown in [Figure 4-25](#).

Create Fault Lifecycle Policy

Clearing Interval (s):

Retention Interval (s):

Soaking Interval (s):

Figure 4-25 Entering the Desired Timer Values

Note

When specifying timer intervals in a lifecycle policy, remember that each type of monitoring policy applies to a different set of objects within the object hierarchy.

Squelching Faults and Changing Fault Severity

There are times when a specific fault generated in an environment may be reported incorrectly. Sometimes false positives result from software defects; other times, there may be certain faults a company wants to ignore due to certain conditions within the network. The process of suppressing, or **squelching**, faults with a specific fault code

helps reduce the noise from a monitoring perspective and allows a company to focus on the faults that really matter.



To squelch a fault, navigate to the Faults view, right-click the fault that should no longer be reported, and select Ignore Fault. Then, in the Ignore Fault window, confirm that the fault code shown should indeed be suppressed, as shown in [Figure 4-26](#). Note that the confirmation window also shows the monitoring policy that will be modified to squelch the fault code.

A screenshot of the Juniper Network Platform interface. The top navigation bar includes tabs for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below this is a secondary navigation bar with links for QuickStart, Dashboard, Controllers, System Settings, Smart Licensing, Faults, Config Zones, Events, Audit Log, and Active Sessions. The main content area is titled "Faults". A table lists various faults with columns for Severity, Domain, Type, Code, Count, Cause, and Sample Fault Description. One row is highlighted with a blue background, corresponding to the fault being ignored. A modal dialog box is open over the table, centered on the selected row. The dialog has a light blue header with an information icon and the text "Ignore Fault". The body of the dialog contains a message: "The system will ignore any fault with code F1940. In order to raise these faults again, you must go to the Monitoring policy below and edit the Fault Severity Assignment policies. This change will apply to all objects referencing the below Monitoring policy." Below this message is a section labeled "Affected Monitoring Policy" with the word "default" and a link to "Fabric > Fabric Policies > Policies > Monitoring > Common Policy > Fault Severity Assignment Policies". At the bottom of the dialog are two buttons: "Cancel" and "Ignore Fault".

Severity	Domain	Type	Code	Count	Cause	Sample Fault Description
Info	Infra	Operational	Ignored Fault	1	Info	urs when a ntp configuration on a controller has problem...
Warning	Access	Operational			Info	urs when the operational state of the ipv4 Nexthop is down...
Warning	External	Config			Info	ised when product license evaluation period (90 days) ...
Info	Infra	Operational			Info	urs when the Power Supply Unit is in fail state
Warning	Infra	Operational			Info	urs when a physical interface on a controller is in the line...
Warning	Infra	Operational			Info	ised when the Policy Manager is not able to assign route...
Info	Tenant	Config			Info	urs when an End Point Group is incompletely or incorre...
Warning	Infra	Config			Info	urs when failed EPg or Vlan field is empty for I2NodeAu...
Warning	Infra	Operational			Info	urs when a ntp configuration on a controller has problem...
Info	Infra	Operational			Info	ssing alert for class I2ingrBytes5min, property dropRate
Info	Infra	Config			Info	used by a hardware programming failure
Info	Infra	Operational			Info	ssing alert for class I2ingrPkts5min, property dropRate
Info	External	Config			Info	This fault is raised when API Controller product is not registered wi...

Figure 4-26 Squelching a Fault from the Faults View

If you decide that the fault code should no longer be squelched, you can navigate to the affected monitoring

policy and delete the fault code from the Fault Severity Assignment Policies folder, as shown in [Figure 4-27](#).

The screenshot shows the Cisco Fabric Manager web interface. The top navigation bar includes tabs for System, Tenants, Fabric (which is selected), Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below the navigation bar is a secondary header with Inventory, Fabric Policies (selected), and Access Policies. On the left, a sidebar titled 'Policies' contains a tree view of policy types: Quick Start, Pods, Switches, Modules, Interfaces, Policies (Pod, Switch, Interface, Global, Monitoring (Fabric Node Controls, Common Policy, Health Score Evaluation Policies, Stats Collection Policies, Syslog Message Policies, Callhome/Smart Callhome/SNMP/Syslog/TACACS, Event Severity Assignment Policies, Fault Severity Assignment Policies, Fault Lifecycle Policy)), default (Stats Collection Policies, Power Control Policies), and others like Stats and Power Control. The 'Fault Severity Assignment Policies' node under 'Monitoring' is expanded. The main content area is titled 'Fault Severity Assignment Policies' and displays a table with one row:

Code	Initial Severity	Target Severity	Description
F1940	squelched	inherit	

Figure 4-27 Reversing a Fault Suppression

To squelch a fault within the scope of a specific monitoring policy, you can navigate to the monitoring policy in question, open the Fault Severity Assignment Policies folder, open the pull-down menu on the right, and select Modify Fault Severity Assignment Policies, as shown in [Figure 4-28](#).

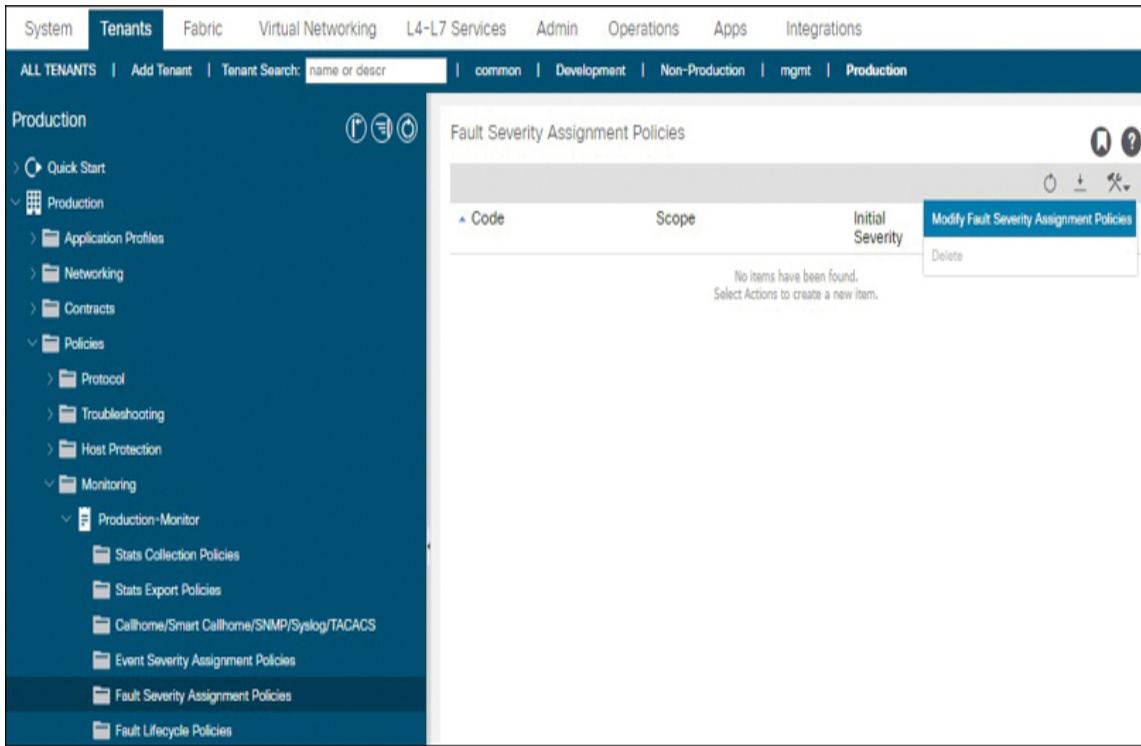


Figure 4-28 Squelching a Fault Code Under a Specific Monitoring Policy

Figure 4-29 shows all faults with fault code F2409 pertinent to Bridge Domain objects being squelched. The monitoring policy under modification in this case is the default monitoring policy in the common tenant. Notice that for squelching to take place, the initial severity of the fault needs to be set to squelched, but the target severity does not need to be modified from inherit.

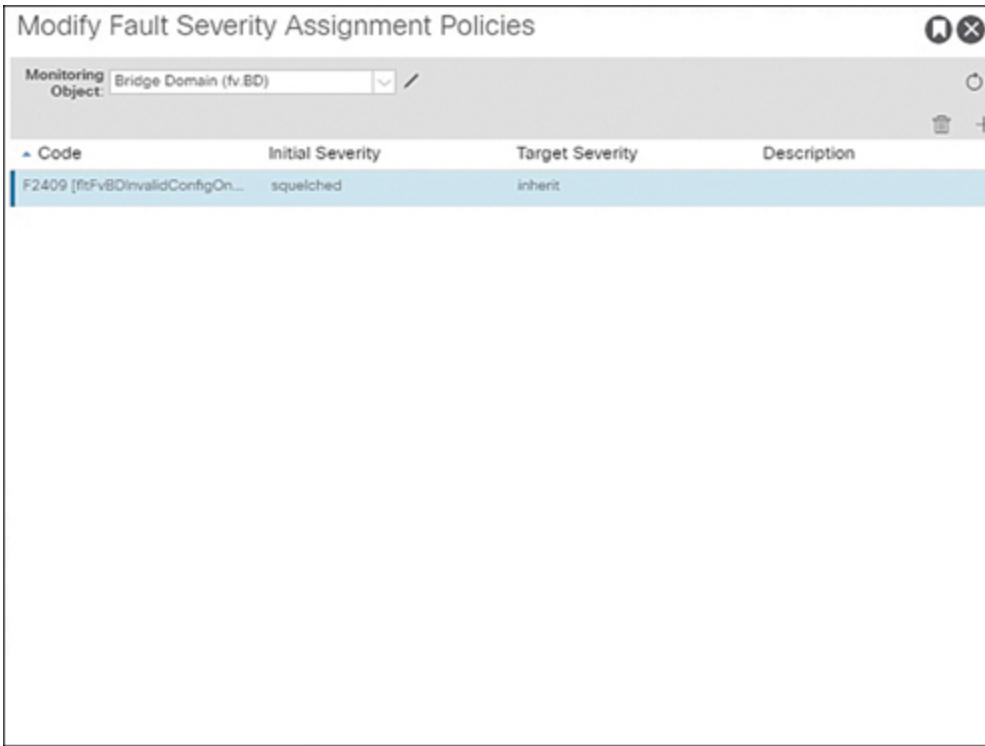


Figure 4-29 *Setting the Initial Severity of an Object to Squelched*

Remember that the default monitoring policy in the common tenant also serves as the default monitoring policy for all other tenants. Be aware of the scope of impact for any monitoring policy changes.

This last example involves fault suppression and also introduces fault severity modification. What else can fault severity modification be used for? Let's say that a fault of little significance shows up in an environment with a higher severity than what the monitoring team deems reasonable. In such a case, the target severity for the fault code can be lowered slightly to reflect the desired severity.

Understanding Health Scores

By using **health scores**, an organization can evaluate and report on the health and operation of managed objects,

switches, tenants, pods, or the entire ACI fabric. By associating a weight with each fault, ACI provides a means for allocating health scores to objects. An object whose children and associated objects are not impacted by faults has a health score of 100. As faults occur, the health score of an object diminishes until it trends toward 0. With the resolution of all related faults, the health score returns to 100.

Key Topic

Figure 4-30 shows three health score panels in the System dashboard. The System Health panel presents the health score of the entire fabric over time. The Nodes panel provides a view of all switches in the fabric whose health scores are less than 100. The Tenants panel lists all tenants whose health scores are less than 100.

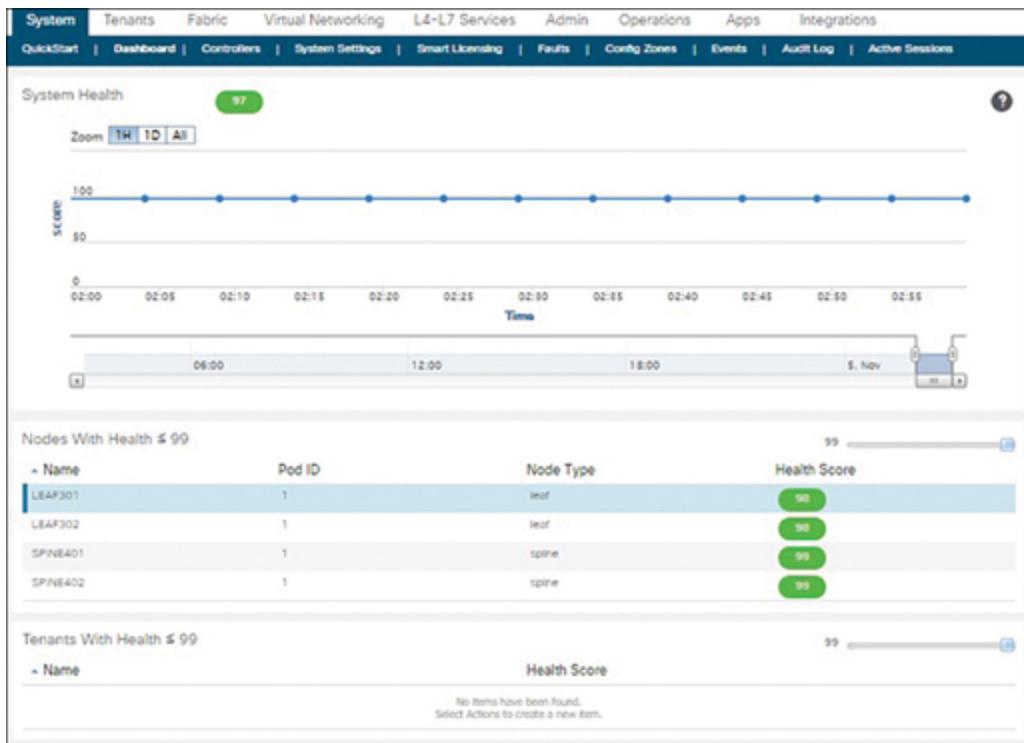


Figure 4-30 View of Health Score Panels from the System Dashboard

The highlighted leaf switch named LEAF301 in Figure 4-30 has a health score of 98. Double-clicking a node or tenant from the health panels opens the object in the GUI. By navigating to the Health tab of the object in question, as shown in Figure 4-31, you can drill down into the details of why the object has a degraded health score. In this case, it might make the most sense to begin the troubleshooting process by investigating the child objects that are impacted by faults and that have the most degraded health scores.

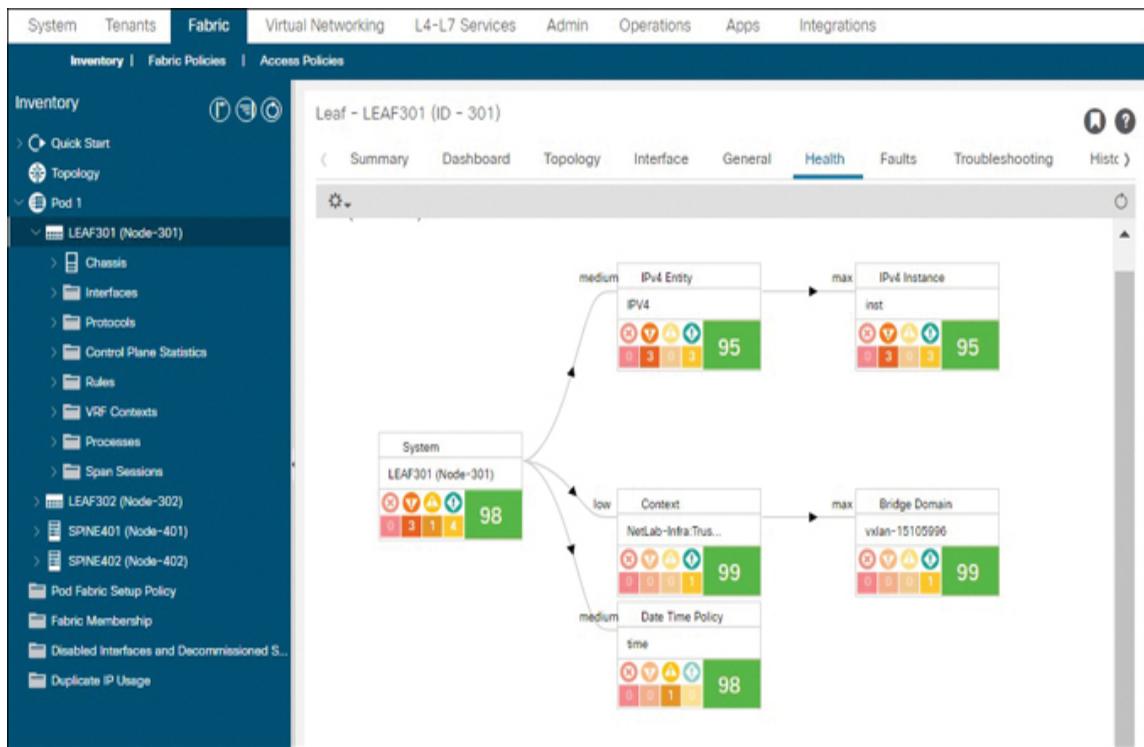


Figure 4-31 The Health Tab for an Object

Note

The companies with the best operations teams make a point of addressing faults and identifying the underlying causes of health score degradation in ACI fabrics as

much as possible. Health score analysis can be used not only for reactive monitoring but as an ideal tool for proactive monitoring. If health degradation occurs frequently, for example, it may point to issues such as packet loss or oversubscription, knowledge of which can greatly assist in capacity planning and proactive mitigation of performance issues.

To modify the weights or percentage of health degradation associated with each fault of a given severity, navigate to the common monitoring policy, as shown in [Figure 4-32](#), and edit the Health Score Evaluation policy. [Figure 4-32](#) shows the default values associated with each fault severity in ACI Release 4.2. To remove acknowledged faults from the health score calculation, you can enable the Ignore Acknowledged Faults option.

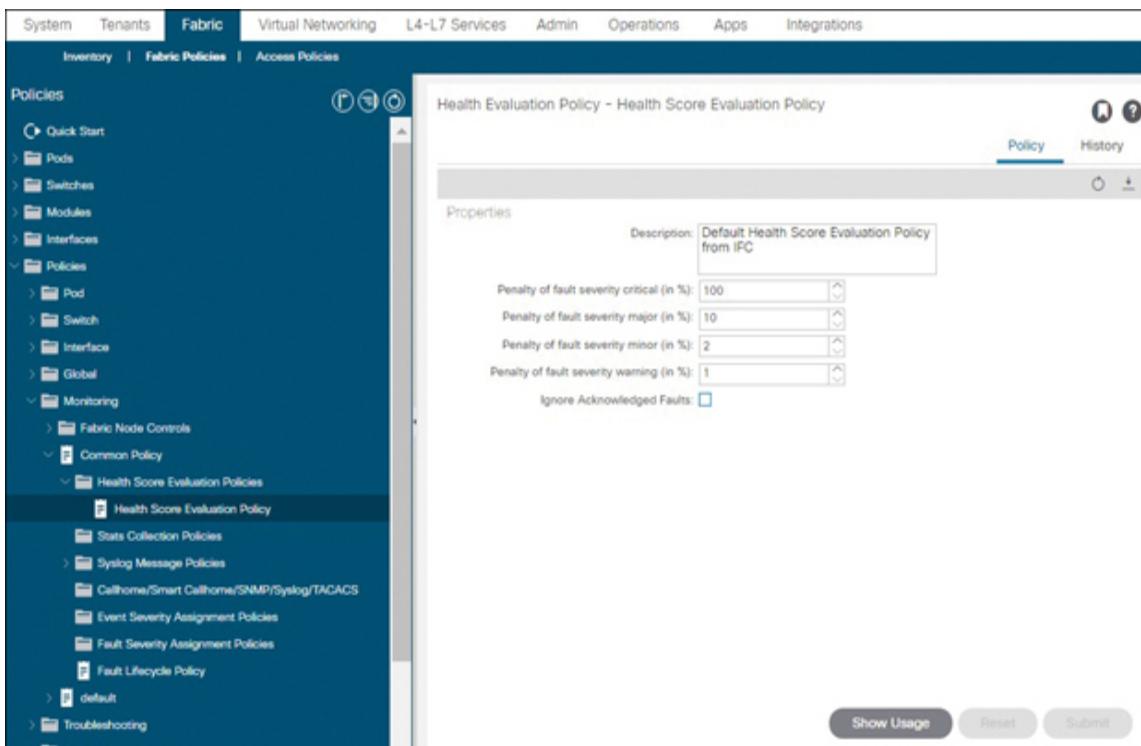


Figure 4-32 Modifying Health Score Calculation Weights for Each Fault Severity

Understanding Events

Event records are objects that are created by a system to log the occurrence of a specific condition that might be of interest to ACI administrators. An event record contains the fully qualified domain name (FQDN) of the affected object, a timestamp, and a description of the condition. Examples of events logged by ACI include link-state transitions, starting and stopping of protocols, and detection of new hardware components. Event records are never modified after creation and are deleted only when their number exceeds the maximum value specified in the event retention policy.



To view a list of events in a system, navigate to System and click on Events, as shown in [Figure 4-33](#).

A screenshot of the Cisco ACI System interface. The top navigation bar includes tabs for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. Below this is a secondary navigation bar with links for QuickStart, Dashboard, Controllers, System Settings, Smart Licensing, Faults, Config Zones, Events (which is highlighted in blue), Audit Log, and Active Sessions. The main content area is titled "Events". It features a table with columns: Severity, Affected Object, Code, Cause, Creation Time, and Description. The "All Events" tab is selected. The table lists ten entries, each with a severity icon (blue exclamation mark) and a timestamp. The descriptions are truncated versions of the full event logs. At the bottom, there are pagination controls (Page 1 of 2632), an "Objects Per Page" dropdown set to 100, and a status message "Displaying Objects 1 - 100 Of 263147".

Severity	Affected Object	Code	Cause	Creation Time	Description
!	uni/usersessexact/session-Vd3WKJe_Scktp3AYcz0wsQ==	E4215060	transition	2019-11-06T01:56:55...	ActiveUserSession Vd3WKJe_Scktp3AYcz0wsQ== deleted
!	uni/usersessexact/session-NSabt+9LQ4iJaWlQQaBow==	E4215058	transition	2019-11-06T01:52:50...	ActiveUserSession NSabt+9LQ4iJaWlQQaBow== created
!	uni/usersessexact/session-bfT9iB95R7yzg1s3gxTWdQ==	E4215058	transition	2019-11-06T01:52:50...	ActiveUserSession bfT9iB95R7yzg1s3gxTWdQ== created
!	uni/usersessexact/session-nsKvyO_cSjee7000p6P_OA==	E4215058	transition	2019-11-06T01:52:49...	ActiveUserSession nsKvyO_cSjee7000p6P_OA== created
!	uni/usersessexact/session-HzYohglpRFqvhfI4EOvLmg==	E4215060	transition	2019-11-06T01:50:54...	ActiveUserSession HzYohglpRFqvhfI4EOvLmg== deleted
!	uni/usersessexact/session-Ffb5_EN1TnGYE9lnaxCZ_Q==	E4215060	transition	2019-11-06T01:50:54...	ActiveUserSession Ffb5_EN1TnGYE9lnaxCZ_Q== deleted
!	uni/usersessexact/session-Kqmdl+vDSkKZzsSVjMxljg==	E4215060	transition	2019-11-06T01:50:54...	ActiveUserSession Kqmdl+vDSkKZzsSVjMxljg== deleted
!	uni/usersessexact/session-sWHYPv6KTgKsZDk_Y_J9Rg==	E4215058	transition	2019-11-06T01:46:56...	ActiveUserSession sWHYPv6KTgKsZDk_Y_J9Rg== created
!	uni/usersessexact/session-Vd3WKJe_Scktp3AYcz0wsQ==	E4215058	transition	2019-11-06T01:46:50...	ActiveUserSession Vd3WKJe_Scktp3AYcz0wsQ== created
!	uni/usersessexact/session-ZDifTEhRSFuOOASCFCfUYg==	E4215058	transition	2019-11-06T01:46:50...	ActiveUserSession ZDifTEhRSFuOOASCFCfUYg== created

Figure 4-33 Viewing Events Under the System Menu

When using event records to troubleshoot specific issues, it is usually most beneficial to navigate to the object most relevant to the problem at hand. Say that an administrator has been asked to troubleshoot a server outage. She navigates to the event record view for the switch port that connects to the server and finds that the switch interface has transitioned out of the up state. [Figure 4-34](#) shows the port Events view under the object History tab. Oftentimes, the Description column provides a hint about the cause of the event. In this case, the description “Physif eth 1/2 modified” suggests that a user may have disabled the interface intentionally, but the event record provides no indication which user disabled the port.

The screenshot displays a network management interface with a top navigation bar containing tabs for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. The Fabric tab is selected. Below the navigation bar is a secondary header with Inventory, Fabric Policies, and Access Policies. The main content area is titled "Layer 1 Physical Interface Configuration - 102/eth1/2". A navigation bar within this section includes Deployed EPGs, VLANs, Stats, QoS Stats, Error Counters, Health, Faults, Troubleshooting, History (which is selected), Events (underlined), and Health. The Events tab is active, showing a table of eight event records. The columns are Severity, Affected Object, Code, Cause, Creation Time, and Description. The events are as follows:

Severity	Affected Object	Code	Cause	Creation Time	Description
!	topology/pod-1/node-102/sys/phys-[eth1/2]	E4208843	transition	2019-11-06T03:30:48.778+0...	Physif eth1/2 modified
!	topology/pod-1/node-102/sys/phys-[eth1/2]/phys	E4215670	port-up	2019-11-06T03:30:43.314+0...	Port is up
!	topology/pod-1/node-102/sys/phys-[eth1/2]/phys	E4205125	port-up	2019-11-06T03:30:43.313+0...	Port is up
!	topology/pod-1/node-102/sys/phys-[eth1/2]	E4208843	transition	2019-11-06T03:30:42.544+0...	Physif eth1/2 modified
!	topology/pod-1/node-102/sys/phys-[eth1/2]	E4208843	transition	2019-11-06T03:30:36.622+0...	Physif eth1/2 modified
!	topology/pod-1/node-102/sys/phys-[eth1/2]/phys	E4205126	port-down	2019-11-06T03:30:36.564+0...	Port is down. Reason: adminCfgChng
!	topology/pod-1/node-102/sys/phys-[eth1/2]/phys	E4215671	port-down	2019-11-06T03:30:36.564+0...	Port is down. Reason: adminCfgChng
!	topology/pod-1/node-102/sys/phys-[eth1/2]	E4208843	transition	2019-11-06T03:30:36.545+0...	Physif eth1/2 modified
!	topology/pod-1/node-102/sys/phys-[eth1/2]	E4208843	transition	2019-11-05T22:18:18.309+0...	Physif eth1/2 modified

Figure 4-34 Using Event Records as a Troubleshooting Tool

Squelching Events

Events can be squelched in a similar way to faults. The easiest way to squelch events of a specific event code is to right-click an event of a specific type and select Ignore Event. This method of squelching events was introduced in ACI Release 4.2(1).

To manually squelch an event of a particular event code, navigate to the pertinent monitoring class object, click Event Severity Assignment Policies, select an object from the Monitoring Object pull-down, select the event code of interest, and set Severity to Squelched, as shown in [Figure 4-35](#).

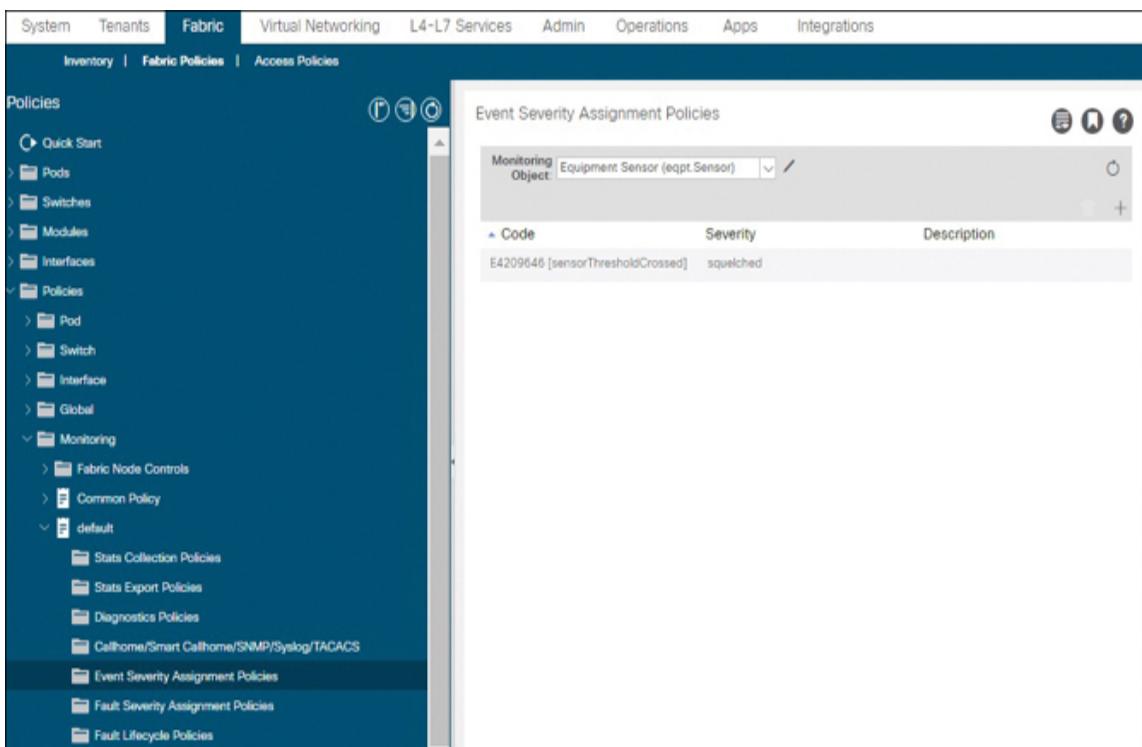


Figure 4-35 *Squelching Events of a Specific Event Code Under Monitoring Policies*

To un-squelch the event code, delete the squelch entry from the Event Severity Assignment policy.

Understanding Audit Logs

Audit logs are records of user actions in ACI, such as logins, logouts, object creations, object deletions, and any other configuration changes (object attribute changes).



Figure 4-34, earlier in this chapter, shows that port 1/2 on a leaf switch was likely shut down intentionally by another ACI user. You can use the audit log to try to find more data on this event. Figure 4-36 shows how you can navigate to **System > Audit Log**. After exploring the audit logs, you might find that the admin user initiated the configuration change that led to the port shutdown. After following up with the one user who has access to the admin user credentials, you might find that the admin user shut down the wrong port. You could then reenable the port to restore service.

System						Tenants	Fabric	Virtual Networking	L4-L7 Services	Admin	Operations	Apps	Integrations
QuickStart Dashboard Controllers System Settings Smart Licensing Faults Config Zones Events Audit Log Active Sessions													
Audit Log													
Time Stamp	ID	User	Action	Affected Object	Description								
2019-11-06T03:56:28.554+00:00	4294977326	admin	creation	uni/fabric/monfab-default/tarfab-eqptSensor/esevp-E4209646	SevAsnP E4209646 created								
2019-11-06T03:55:42.704+00:00	4294977319	admin	deletion	uni/fabric/monfab-Blah2/coll-1year	HierColl 1 Year deleted								
2019-11-06T03:55:42.704+00:00	4294977320	admin	deletion	uni/fabric/monfab-Blah2/coll-15min	HierColl 15 Minute deleted								
2019-11-06T03:55:42.704+00:00	4294977321	admin	deletion	uni/fabric/monfab-Blah2/coll-1h	HierColl 1 Hour deleted								
2019-11-06T03:55:42.704+00:00	4294977322	admin	deletion	uni/fabric/monfab-Blah2/coll-1w	HierColl 1 Week deleted								
2019-11-06T03:55:42.704+00:00	4294977323	admin	deletion	uni/fabric/monfab-Blah2/coll-1mo	HierColl 1 Month deleted								
2019-11-06T03:55:42.704+00:00	4294977324	admin	deletion	uni/fabric/monfab-Blah2/coll-5min	HierColl 5 Minute deleted								
2019-11-06T03:55:42.704+00:00	4294977325	admin	deletion	uni/fabric/monfab-Blah2/coll-1qtr	HierColl 1 Quarter deleted								
2019-11-06T03:55:42.703+00:00	4294977317	admin	deletion	uni/fabric/monfab-Blah2	FabricPol Blah2 deleted								
2019-11-06T03:55:42.703+00:00	4294977318	admin	deletion	uni/fabric/monfab-Blah2/coll-1d	HierColl 1 Day deleted								
2019-11-06T03:30:05.081+00:00	4294977316	admin	deletion	uni/fabric/cutofsvc/rsosPath-[topology/pod-1/paths-102/pathep-[eth1/2]]	RsQoSPath topology/pod-1/paths-102/pathep-[eth1/2] deleted								
2019-11-06T03:29:59.078+00:00	4294977315	admin	creation	uni/fabric/cutofsvc/rsosPath-[topology/pod-1/paths-102/pathep-[eth1/2]]	RsQoSPath topology/pod-1/paths-102/pathep-[eth1/2] created								

Figure 4-36 Reviewing the Audit Logs

There are a lot of objects in ACI that have a History menu, under which audit logs for the object can be accessed. When troubleshooting faults pertinent to a specific object, it is sometimes a good idea to see if any configuration changes were made to the object. The ability to quickly review audit logs helps troubleshoot issues faster.

Note

The class aaaSessionLR represents fabric logins and logouts. The class aaaModLR represents a configuration change within the fabric. Use the commands **moquery -c aaaSessionLR** and **moquery -c aaaModLR** to query the object model and understand user actions.

Exam Preparation Tasks

As mentioned in the section “How to Use This Book” in the Introduction, you have a couple of choices for exam preparation: [Chapter 17](#), “Final Preparation,” and the exam simulation questions on the companion website.

Review All Key Topics

Review the most important topics in this chapter, noted with the Key Topic icon in the outer margin of the page. [Table 4-8](#) lists these key topics and the page number on which each is found.



Table 4-8 Key Topics for [Chapter 4](#)

Key Topic Element	Description	Page Number
Paragraph	Explains the MIM and Policy Universe	105
Paragraph	Defines Visore	108
Paragraph	Defines distinguished name	109
Paragraph	Defines class	109

Key Topic Element	Description	Page Number
Paragraph	Defines MO	109
Paragraph	Describes MOQuery and most rudimentary options	110
Paragraph	Describes faults	111
Table 4-3	Lists fault severity levels users may see in the Faults page	112
Table 4-4	Lists fault types	113
Table 4-5	Lists fault lifecycle phases	114
Table 4-6	Lists fault lifecycle intervals	115
Paragraph	Provides an example of an MO attribute and describes monPolDn	116

Key Topic Element	Description	Page Number
Table 4-7	Lists classes of monitoring policies	118
Paragraph	Explains the use case and definition of fault squelching	121
Paragraph	Describes health scores	124
Paragraph	Describes event records	126
Paragraph	Describes audit logs	127

Complete Tables and Lists from Memory

Print a copy of [Appendix C, “Memory Tables”](#) (found on the companion website), or at least the section for this chapter, and complete the tables and lists from memory. [Appendix D, “Memory Tables Answer Key”](#) (also on the companion website), includes completed tables and lists you can use to check your work.

Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

Management Information Model (MIM)

Policy Universe

user tenant

fabric policy

access policy

managed object (MO)

Visore

MOQuery

distinguished name (DN)

fault

monPolDn

fault lifecycle policy

squelching

health scores

event record

audit log

Part II: ACI Fundamentals

Chapter 5

Tenant Building Blocks

This chapter covers the following topics:

Understanding the Basic Objects in Tenants: This section describes the key logical constructs in tenants, including bridge domains and EPGs.

Contract Security Enforcement Basics: This section details how ACI uses contracts, subjects, filters, and filter entries to enforce whitelisting.

Objects Enabling Connectivity Outside the Fabric: This section describes how L3Outs and external EPGs fit in the bigger picture of tenants.

Tenant Hierarchy Review: This section covers the relationships between tenant objects.

This chapter covers the following exam topics:

- 1.6 Implement ACI logical constructs
 - 1.6.a tenant
 - 1.6.b application profile
 - 1.6.c VRF
 - 1.6.d bridge domain (unicast routing, Layer 2 unknown hardware proxy, ARP flooding)

- 1.6.e endpoint groups (EPG)
- 1.6.f contracts (filter, provider, consumer, reverse port filter, VRF enforced)

Because ACI functions are based on objects, it is reasonable to expect that a book introducing ACI as a multitenant solution would include detailed coverage of the theory around objects that make up tenants. This chapter begins with an overview of key tenant constructs that all ACI engineers need to know. It provides a basic understanding of how contracts enforce security in ACI. Because ACI needs to also enable communication with the outside world, this chapter also discusses the role of tenant L3Outs and related objects.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 5-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Questions.”](#)

Table 5-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding the Basic Objects in Tenants	1-5

Contract Security Enforcement Basics	6-8
Objects Enabling Connectivity Outside the Fabric	9
Tenant Hierarchy Review	10

Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security

- 1.** Which tenant does ACI use to push configurations to switches in-band?
 - a.** Mgmt
 - b.** User
 - c.** Infra
 - d.** Common

- 2.** Which tenant allows deployment of a shared L3Out?
 - a.** Mgmt

- b.** User
 - c.** Infra
 - d.** Common
- 3.** Which of the following most accurately describes an EPG?
- a.** An EPG defines a broadcast domain in ACI.
 - b.** An EPG is a logical grouping of IP-based endpoints that reside inside an ACI fabric.
 - c.** An EPG is the equivalent of a VLAN in ACI.
 - d.** An EPG is a logical grouping of endpoints, IP-based or otherwise, that have similar policy-handling requirements and are bound to a single bridge domain.
- 4.** Which of the following statements about application profiles is true?
- a.** EPGs tied to different bridge domains cannot be grouped into a single application profile.
 - b.** An application profile is typically a grouping of EPGs that together form a multitiered application.
 - c.** An application profile is bound to a VRF instance.
 - d.** The function of application profiles is to differentiate between DMZ and inside zones of a firewall.
- 5.** Which of the following commands displays all routes in a VRF instance called DCACI?
- a. `show ip route DCACI:CCNP`**
 - b. `show route DCACI`**
 - c. `show ip route`**
 - d. `show ip route CCNP:DCACI`**

- 6.** Which of the following defines the action that should be taken on interesting traffic?
- a.** Filter
 - b.** Filter entry
 - c.** Subject
 - d.** Contract
- 7.** True or false: There is no way to isolate traffic between endpoints that reside in the same EPG.
- a.** True
 - b.** False
- 8.** An administrator has defined constructs that match traffic based on destination ports 80 and 443, allowing such traffic along with return traffic through the ports. The contract is expected to be applied to communication between a client EPG and a web EPG. How should the contract be applied to the two EPGs to allow the clients to establish communication with the web EPG?
- a.** In the consumer direction on both EPGs
 - b.** In the provider direction on both EPGs
 - c.** In the provider direction on the client EPG and in the consumer direction on the web EPG
 - d.** In the consumer direction on the client EPG and in the provider direction on the web EPG
- 9.** True or false: An external EPG represents endpoints outside ACI and behind an L3Out.
- a.** True
 - b.** False
- 10.** True or false: Large numbers of filters can be created in any given tenant.

- a. True
- b. False

Foundation Topics

Understanding the Basic Objects in Tenants

ACI has multiple tenants enabled out of the box. There is little reason not to deploy multiple user tenants to achieve fault isolation and tighter administrative control. However, to fully leverage ACI multitenancy, you must first master the tenant hierarchy.

In true multitenancy environments where roles are heavily delineated, tenant policies are typically configured by a user who has been assigned either the tenant-admin role or a role with similar privileges. (For more on implementation of roles, see [Chapter 15, “Implementing AAA and RBAC.”](#))

Tenants



An ACI **tenant** is a secure and exclusive virtual computing environment that forms a unit of isolation from a policy perspective but does not represent a private network.

If you investigate further into use cases for tenants in the real world, you will find that tenants are often deployed in order to achieve these two technical controls:

- **Administrative separation:** When a business acquires other entities and needs to allow outside administrators access into its data centers, tenants are often used as a unit of administrative separation. This is accomplished through role-based access control (RBAC). Other instances where administrative separation may be important are when business units or application owners want to be involved in the process of defining network and security policies for applications. In this case, each relevant business unit can be provided its own tenants, or a tenant can be defined and dedicated to a specific application. Another instance in which administrative separation is vital is in service provider environments, where customers sometimes have access and visibility into the endpoints and systems they own.
- **Configuration fault isolation:** An application is a collection of tightly integrated endpoints that need to communicate with one another to achieve a particular business objective. Some applications have low business relevance and some have high business relevance. The networking, security, and QoS handling required for applications are defined in tenants. A hospital, for example, will likely consider its electronic medical record system to be business critical, with very well-defined dependencies, and may want any network or security policy changes around such an environment to be bound by change control. In such a case, it might make sense to place such an application and its dependencies in its own tenant. The same hospital may see a host of other applications as having very little business relevance and may therefore lump such applications into another tenant. The idea here is that configuration changes made in one tenant should have

very limited or no impact on endpoints and applications in other tenants.

Figure 5-1 shows how you can navigate to the Tenants menu in the APIC GUI and execute the tenant creation wizard by clicking Add Tenant.

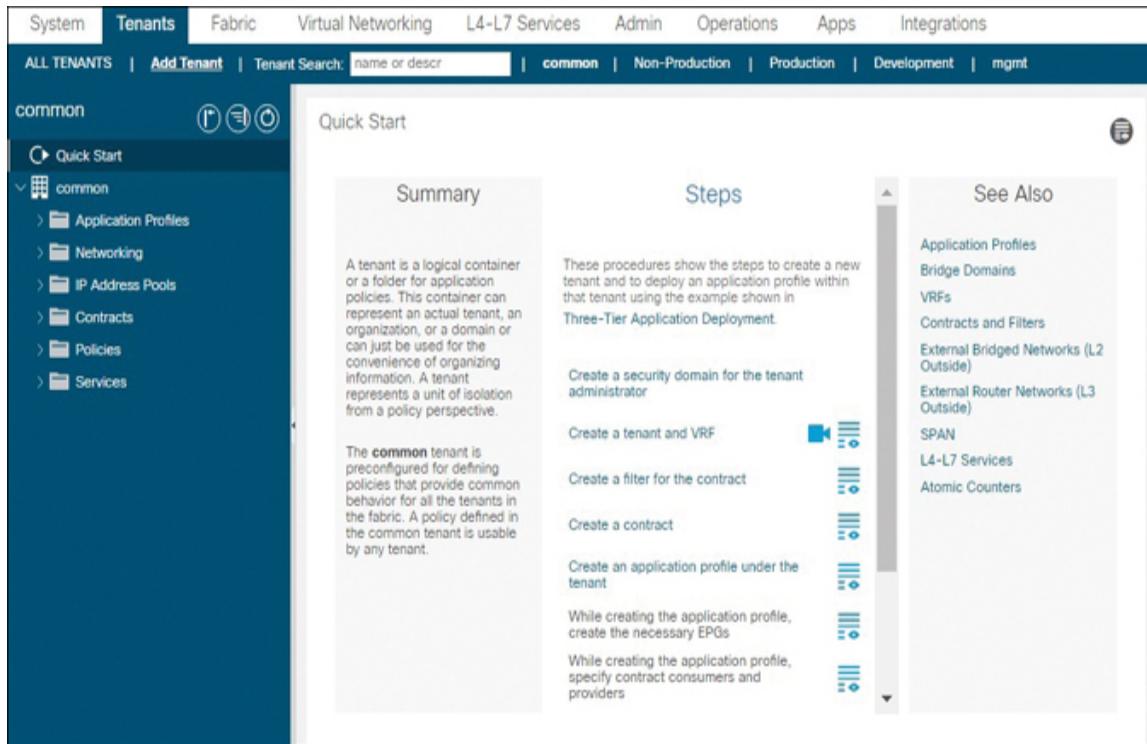


Figure 5-1 Navigating to the Tenants Menu and Clicking Add Tenant

In the Create Tenant wizard, you can enter the name of the desired tenant and click Submit to create the tenant, as shown in **Figure 5-2**. Note in this figure that one of the items you can create simultaneously while in the Create Tenant wizard is a VRF instance (VRFs are discussed later in this chapter.)

Create Tenant

Name:	DCACI
Alias:	
Description:	optional
Tags:	enter tags separated by comma
GUID:	Provider GUID Account Name
Monitoring Policy:	select a value
Security Domains:	Name Description
VRF Name:	optional
<input checked="" type="checkbox"/> Take me to this tenant when I click finish	
Cancel Submit	

Figure 5-2 Create Tenant Wizard

Note

This chapter does not cover all configuration items depicted in the figures. [Chapters 7, “Implementing Access Policies,” through 15](#) address additional configurations and features that are within the scope of the Implementing Cisco Application Centric Infrastructure DCACI 300-620.

Note

Tenants cannot be nested within each other.

Predefined Tenants in ACI

ACI comes preconfigured with three tenants:

Key Topic

- **Infra:** The infra tenant is for internal communication between ACI switches and APICs in an ACI fabric. When APICs push policy to leaf switches, they are communicating into the infra tenant. Likewise, when leaf and spine switches communicate with one another, they do so in the infra tenant. The infra tenant is the underlay that connects ACI switches together and does not get exposed to the user space (user-created tenants). In essence, the infra tenant has its own private network space and bridge domains. Fabric discovery, image management, and DHCP for fabric functions are all handled within this tenant. Note also that an Application Virtual Switch (AVS) software switch can be considered an extension of an ACI fabric into virtualized infrastructure. When AVS is deployed, it also communicates with other ACI components in the infra tenant.
- **Mgmt:** APICs configure switches in a fabric via the infra tenant, but it is likely that administrators at some point will want APIC GUI access or CLI access to nodes within a fabric to validate that a policy has been pushed or to troubleshoot issues. Administrator SSH access to ACI switches and any contracts limiting communication with switch management IP addresses are configured in the mgmt tenant. Both out-of-band and in-band management options are configured in this tenant.
- **Common:** The common tenant is a special tenant for providing common services to other tenants in an ACI fabric. The common tenant is most beneficial for placement of services that are consumed by multiple tenants. Such services typically include DNS, DHCP, and Active Directory. The common tenant also allows the

creation of shared Layer 3 connections outside the fabric, shared bridge domains, and shared VRF instances.

Note

This section refers to the infra tenant as the underlay in ACI. The term *underlay* can technically be used to refer not just to the tenant itself but also to the protocols that enable interswitch connectivity within the fabric. That said, user traffic typically resides in either user-created tenants or the common tenant. Therefore, user tenants and the common tenant can be considered the overlay in ACI.

VRF Instances



A ***virtual routing and forwarding (VRF)*** instance is a mechanism used to partition a routing table into multiple routing tables for the purpose of enabling Layer 3 segmentation over common hardware. In ACI, each tenant can contain multiple VRF instances.

IP addresses within a VRF need to be unique, or traffic can be black-holed. IP address overlap between different VRFs, on the other hand, is not an issue. Where subnet overlap does exist within ACI VRFs, the overlapping subnets *cannot* be leaked between the VRFs to allow communication.

VRF instances are sometimes also referred to as *private networks, or contexts*.

[Figure 5-3](#) provides a view from within the newly created tenant DCACI. To create a VRF instance, navigate to the tenant in which you intend to create the VRF, open Networking, right-click on VRFs, and select Create VRF.

The screenshot shows the Cisco ACI Management interface. The top navigation bar includes tabs for System, Tenants, Fabric, Virtual Networking, L4-L7 Services, Admin, Operations, Apps, and Integrations. The Tenants tab is selected, showing the 'DCACI' tenant. Below the navigation is a search bar and a breadcrumb trail: ALL TENANTS | Add Tenant | Tenant Search: name or descr | common | DCACI | Development | mgmt | Production. The main content area is titled 'Networking - VRFs'. On the left, a sidebar for 'DCACI' lists 'Quick Start', 'DCACI' (selected), 'Application Profiles', 'Networking' (selected), 'Bridge Domains', 'VRFs' (highlighted with a blue box and labeled 'Create VRF'), 'External Bridged Networks', 'External Routed Networks', 'Dot1Q Tunnels', 'Contracts', 'Policies', and 'Services'. The 'VRFs' section in the main content area displays a table with columns: Name, Alias, Segment, Class ID, Policy Control Enforcement Preference, Policy Control Enforcement Direction, and Description. A message at the bottom of the table says 'No items have been found. Select Actions to create a new item.' At the bottom of the page are pagination controls ('Page 0 of 0'), an 'Objects Per Page' dropdown set to 15, and a link 'No Objects Found'.

Figure 5-3 *Navigating to the Create VRF Wizard*

[Figure 5-4](#) displays the Create VRF wizard, in which you enter the name of the desired VRF and click Finish to create the VRF. Note in this figure that you can create bridge domains simultaneously when creating a VRF. (Bridge domains are covered later in this chapter.)

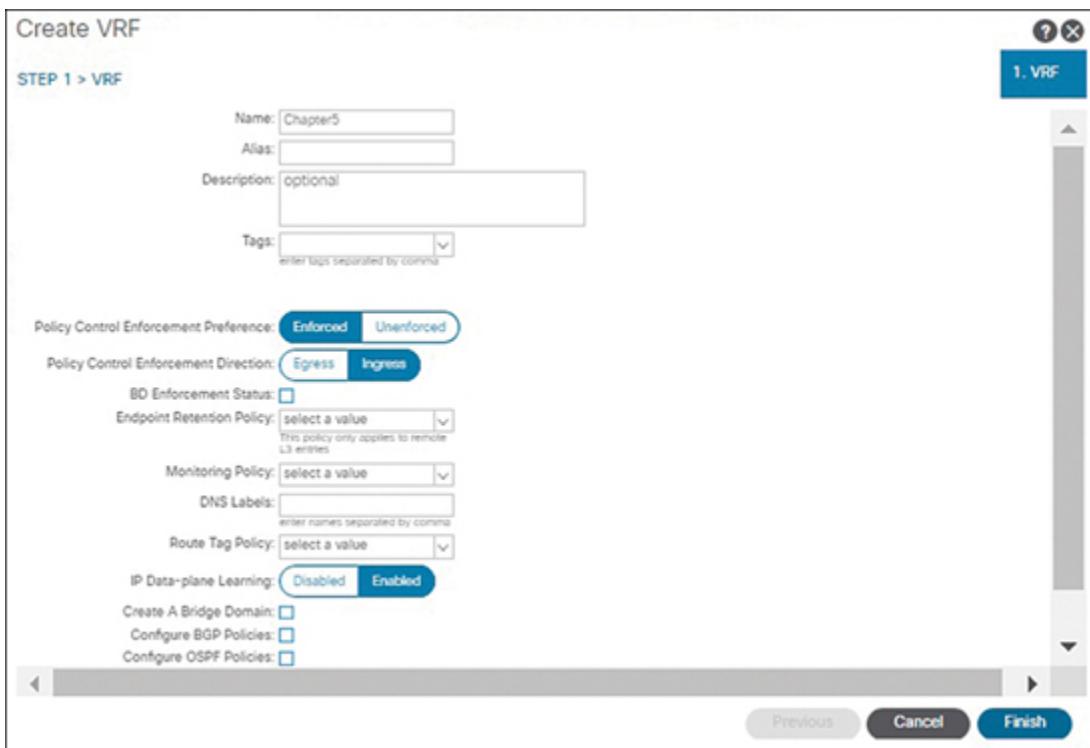


Figure 5-4 Create VRF Wizard

Example 5-1 illustrates that routing tables in ACI find meaning only in the context of VRF instances. There is no concept of a default VRF in ACI. As you can see, the command **show ip route** is invalid in ACI. A reference to a tenant and VRF using the syntax **show ip route vrf { tenant-name:vrf-name }** is required when verifying the routing table of user-created VRFs within ACI. The list of VRFs that have been activated on a leaf and the references needed to pull further output can be identified using the **show vrf** command.

Example 5-1 Routing Table Output in ACI

[Click here to view code image](#)

```
DC1-LEAF101# show ip route
Incorrect command "show ip route"
DC1-LEAF101# show vrf
```

VRF-Name	VRF-I	State	Reason
black-hole	3	Up	--
DCACI:Chapter5	6	Up	--
management	2	Up	--
overlay-1	4	Up	--

```
DC1-LEAF101# show ip route vrf DCACI:Chapter5
```

IP Route Table for VRF "DCACI: Chapter5"

'*' denotes best ucast next-hop

'***' denotes best mcast next-hop

'[x/y]' denotes [preference/metric]

'%<string>' in via output denotes VRF <string>

10.233.52.0/24, ubest/mbest: 1/0, attached, direct, pervasive

*via 10.233.47.66%overlay-1, [1/0], 09w05d, static

10.233.52.1/32, ubest/mbest: 1/0, attached, pervasive

*via 10.233.52.1, vlan12, [0/0], 09w05d, local, local

Note that the subnet and IP addresses shown in [Example 5-1](#) were not created as a result of the VRF instance creation process demonstrated in [Figure 5-3](#) and [Figure 5-4](#).

Bridge Domains (BDs)



Official ACI documentation describes a ***bridge domain (BD)*** as a Layer 2 forwarding construct that is somewhat analogous to a VLAN and has to be associated with a VRF instance.

The official definition, presumably, explains why the term *bridge* has been used in the name of this construct since a

bridge domain is the true boundary of any server-flooded traffic.

Although this definition is technically accurate and must be understood for the purpose of the DCACI 300-620 exam, it is a great source of confusion for newcomers to ACI. So, let's first explore endpoint groups and application profiles and then revisit bridge domains to get a better understanding of the role these two constructs play in the greater picture of ACI.

Endpoint Groups (EPGs)



An ***endpoint group (EPG)*** is a grouping of physical or virtual network endpoints that reside within a single bridge domain and have similar policy requirements. Endpoints within an EPG may be directly or indirectly attached to ACI leaf switches but communicate in some fashion over an ACI fabric. ACI can classify both IP-based and non-IP-based endpoints into EPGs.

Some examples of endpoints that can be classified into EPGs include virtual machines, physical servers, appliance ports, Kubernetes namespaces, and users accessing ACI.

Application Profiles



An ***application profile*** is a container that allows EPGs to be grouped according to their relationship with one another

to simplify configuration and auditing of relevant policies and to enable a level of policy reuse.

Many modern applications contain multiple components (tiers). For instance, an e-commerce application could require one or more web servers, backend database servers, storage, and access to outside resources that enable financial transactions. In ACI deployments, especially if whitelisting is desired, each one of these component types (for example, web servers) would be classified into a separate EPG. An important benefit of organizing interrelated component EPGs of a multitiered application into an application profile container is that the allowed communication between these application tiers can then be easily audited by exploring the resulting application profile topology. Figure 5-5 presents a sample application profile topology comprising a web tier and a database tier rendered by ACI.

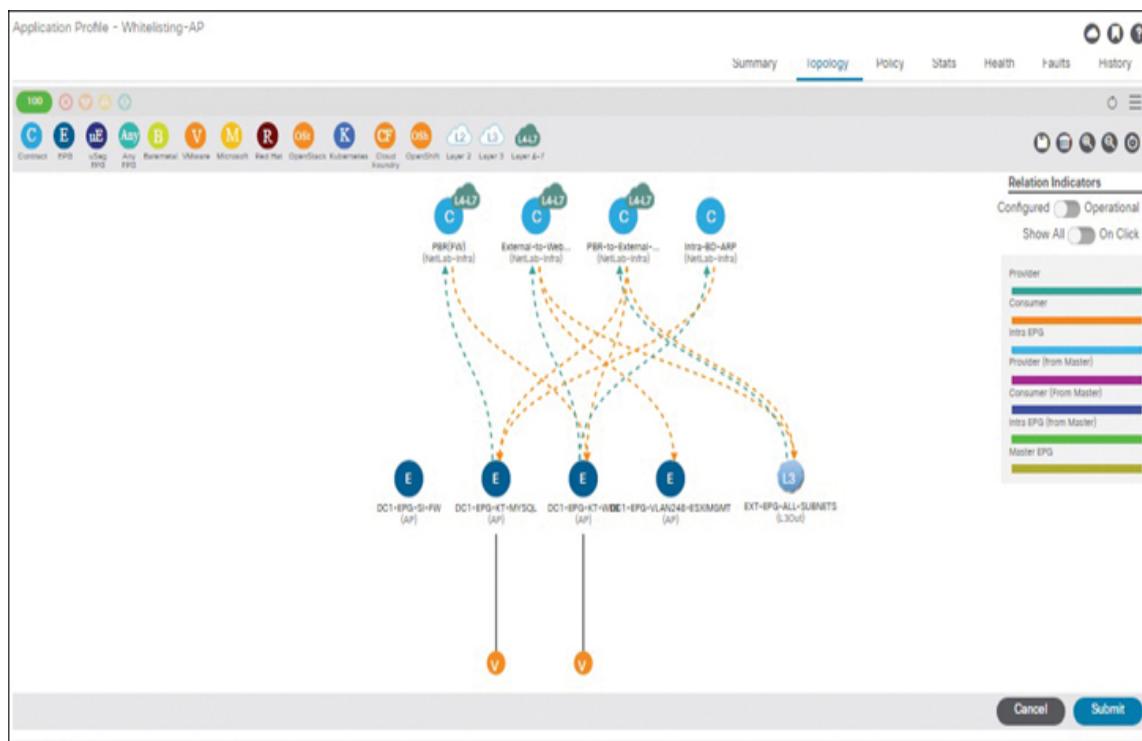


Figure 5-5 An Application Profile Topology

Another advantage of application profiles is that they allow application policy requirements to be modeled to accommodate policy standardization and future reuse. For example, let's say an IT department finds itself spinning up new instances of a very specific multitiered application very often. It understands the communication protocols that should be allowed between various tiers of these standardized deployments because it has already had to implement whitelisting policies for a previous multitiered instance of this same application. By limiting the scope of policies that have already been created to that of an application profile, the IT department can apply the same policies to new instances of this multitiered application without having the various instances of the application communicating with one another. This requires that the new instance of the application be placed in a new application profile. This type of policy reuse can cut down on the time needed to deploy applications and ensures that when policy changes are needed (for example, when a new port needs to be opened between application tiers), they can be applied to all instances at once.

Note

Scope-limiting policies and creating new application profiles for each application instance is not the only way to take advantage of policy reuse for standardized applications in ACI. In many cases with ACI, you can achieve desired business or technical objectives in multiple ways.

EPGs can be organized into application profiles according to one of the following:

- The application they provide, such as a DNS server, a LAMP stack, or SAP

- The function they provide (such as infrastructure)
- Where they are in the structure of the data center (such as the DMZ)
- Any organizing principle that a tenant administrator chooses to use



EPGs that are placed in an application profile do not need to be bound to the same bridge domain. In addition, application profiles are not tied to VRF instances.

The Pain of Designing Around Subnet Boundaries

In traditional data centers, security policy in particular is usually applied at subnet boundaries. Access lists are rarely used to drop traffic flows within traditional data centers, but when they are, they are almost always used for isolated use cases that do not involve very granular control at the individual IP address level.

For example, technical controls such as access lists and route maps may be used in traditional data centers to prevent non-production server traffic from reaching a production server block *if* production and non-production server blocks have very well-defined subnets and no interdependencies. However, it is very unlikely that an organization that uses traditional networking capabilities would leverage its data center network to set up controls and define policy for limiting communications to and from every single server.

Where application-level firewalling is needed for an endpoint or set of endpoints within a traditionally built data center, careful engineering is applied to ensure that traffic is pushed through a firewall. The common traditional solution to a requirement like this may be to build out a new security zone on a firewall and move the default gateway for the subnet in question onto the firewall to guarantee that the firewall has control over traffic flowing into and out of the subnet. This type of solution, shown in [Figure 5-6](#), forces engineers to think a lot about subnet boundaries when designing networks.

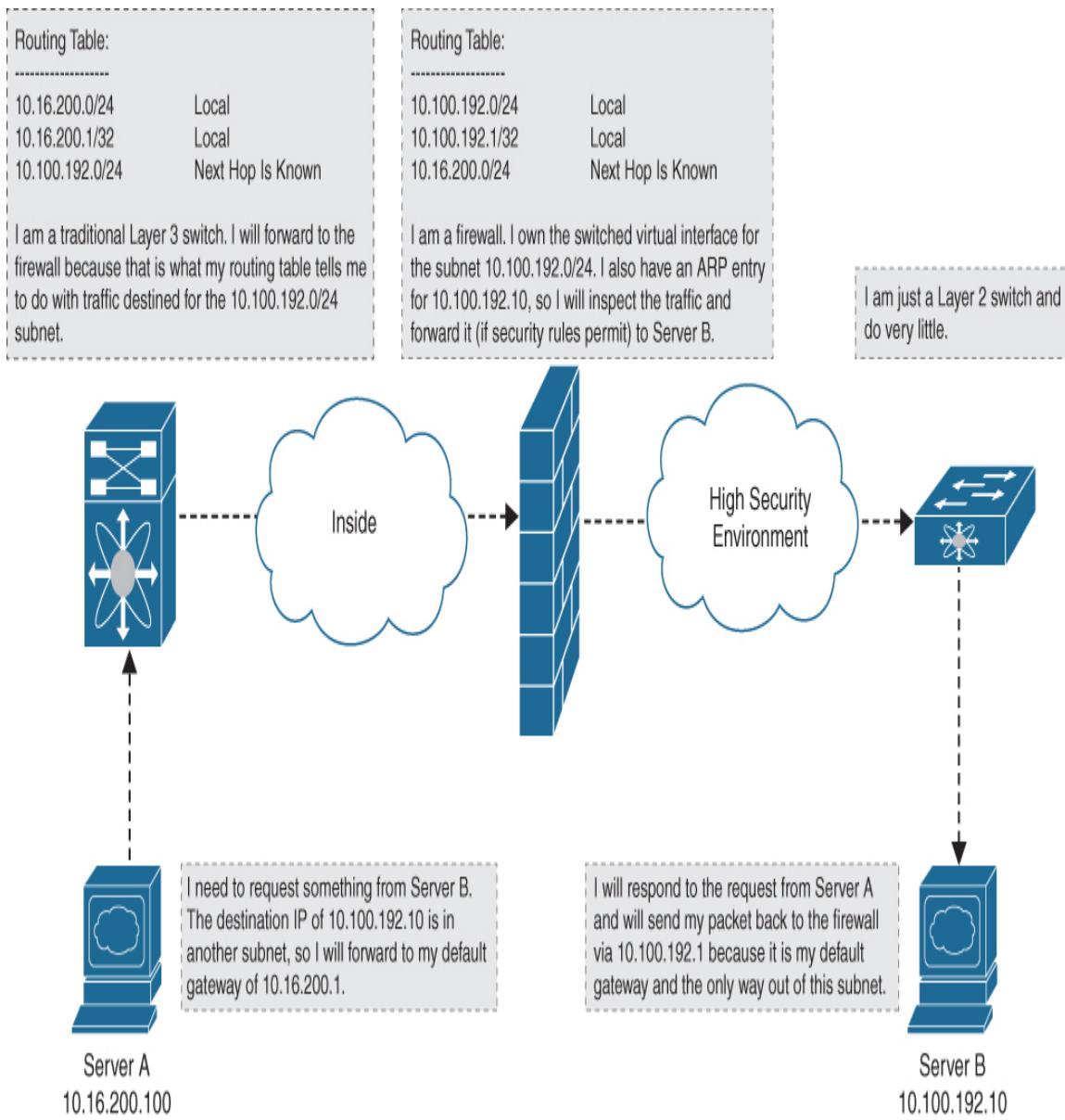


Figure 5-6 Security by Routing Traffic Through Firewalls

Sometimes engineers may decide to enforce security by leveraging a firewall in transparent mode in conjunction with an isolation VLAN. This solution ensures that certain critical endpoints are firewalled off from other endpoints within a subnet and allows for limited policy control within a subnet boundary. [Figure 5-7](#) demonstrates how a transparent firewall attached to a traditional network can be placed between endpoints within a subnet to segment the subnet

into two VLANs (VLAN 100 and 200 in this case) to enforce security policies between the VLANs.

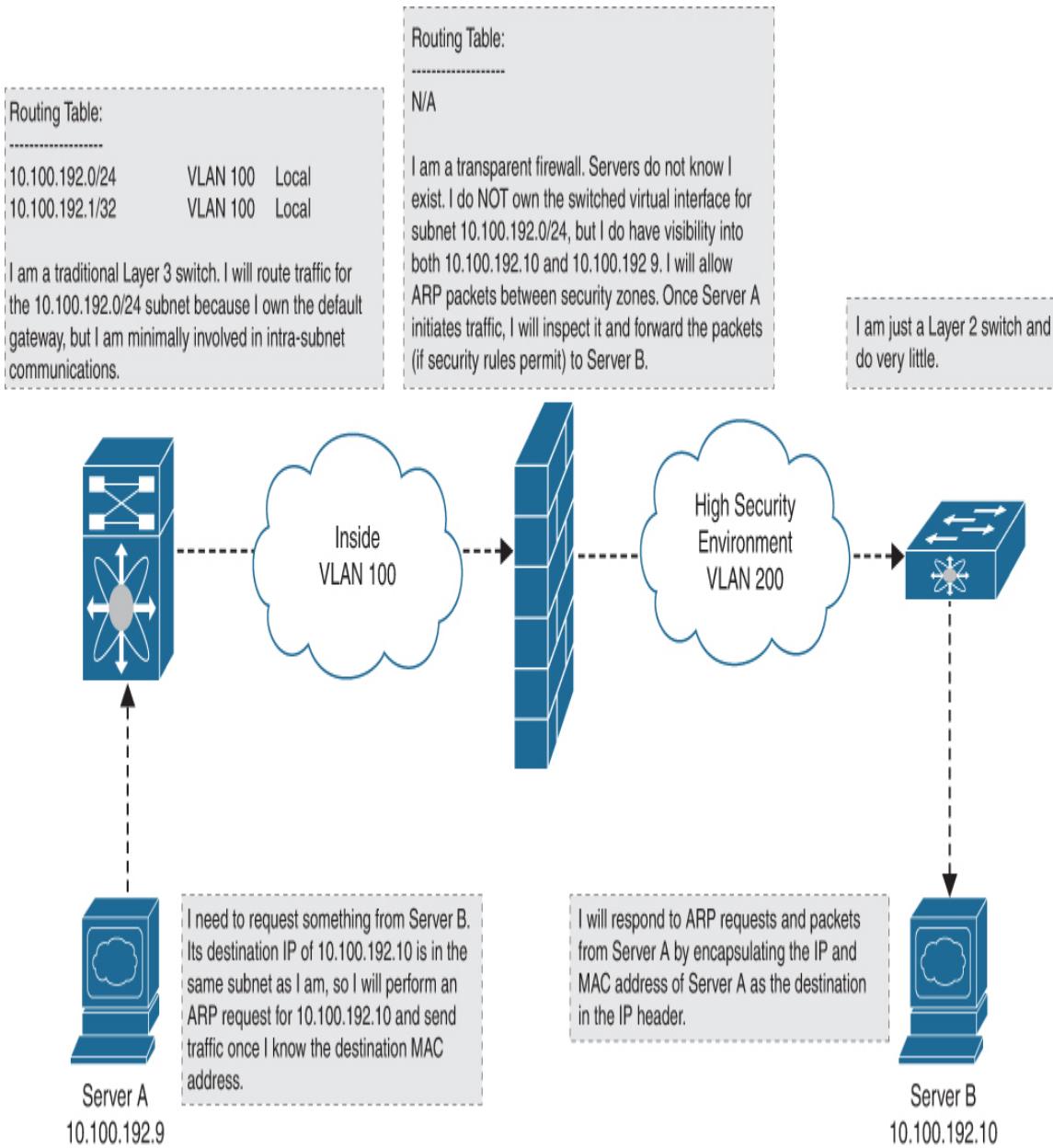


Figure 5-7 Security Through Transparent Bridging via Firewalls

There are several challenges associated with both of these security solutions when implemented using traditional networking capabilities. First, if the servers are already in

production, dropping a firewall into the traffic path after the fact almost always necessitates an outage of the servers that are firewalled off. This is particularly true when segmenting the subnet in which the server resides. Second, granular east-west traffic filtering using these methods is nearly impossible. (For instance, what happens if a subnet needs to be subdivided into 10 sets of servers and security zones?) Finally, even with these methods, there is very little that can be done to specifically direct only the desired traffic through security devices. In other words, engineers may find that it requires a lot more planning to design a solution that sends all traffic from certain servers to firewalls if a required secondary objective were for traffic from these servers to completely bypass said firewalls when the traffic is found to be destined toward backup appliances.

The complexity of enforcing solutions to security challenges using traditional data center networks underscores the basic point that subnet boundaries play an important role in the average data center. Even though security policy has been the main focus in this discussion, the reality is that the challenge and rigidity involved in designing networks with subnet boundaries in mind also extend to other aspects of policy enforcement.

BDs and EPGs in Practice

Unlike traditional networks, ACI breaks the shackles and endless limitations imposed by subnet boundaries to eliminate the need for overengineered designs. It does so by decoupling Layer 3 boundaries from security and forwarding policies.

For the purpose of gaining a fuller picture, let's redefine bridge domains and EPGs based on their practical application.

Key Topic

As a construct that is directly associated with a VRF instance, a bridge domain serves as the subnet boundary for any number of associated EPGs. One or more subnets can be assigned to a bridge domain. General forwarding aspects of the associated subnets—such as whether flooding and multicast are enabled or whether the subnets should be advertised out of an ACI fabric or not—are governed by the bridge domain.

Key Topic

Endpoints that live within a bridge domain subnet need to be associated with an EPG to be able to forward traffic within ACI. An EPG serves as an endpoint identity from a policy perspective. EPGs are the point of security policy enforcement within ACI. Traffic flowing between EPGs can be selectively filtered through the use of contracts. Policies not necessarily related to security, such as QoS, can also be applied at the EPG level. If traffic from a set of endpoints may need to be selectively punted to a firewall or any other stateful services device, a policy-based redirect (PBR) operation can be applied to the EPG to bypass the default forwarding rules. In a sense, therefore, EPG boundaries also have a hand in the application of forwarding policies.

Figure 5-8 demonstrates how ACI decouples policy from forwarding by using bridge domains as the subnet definition point and EPGs as the policy application point.

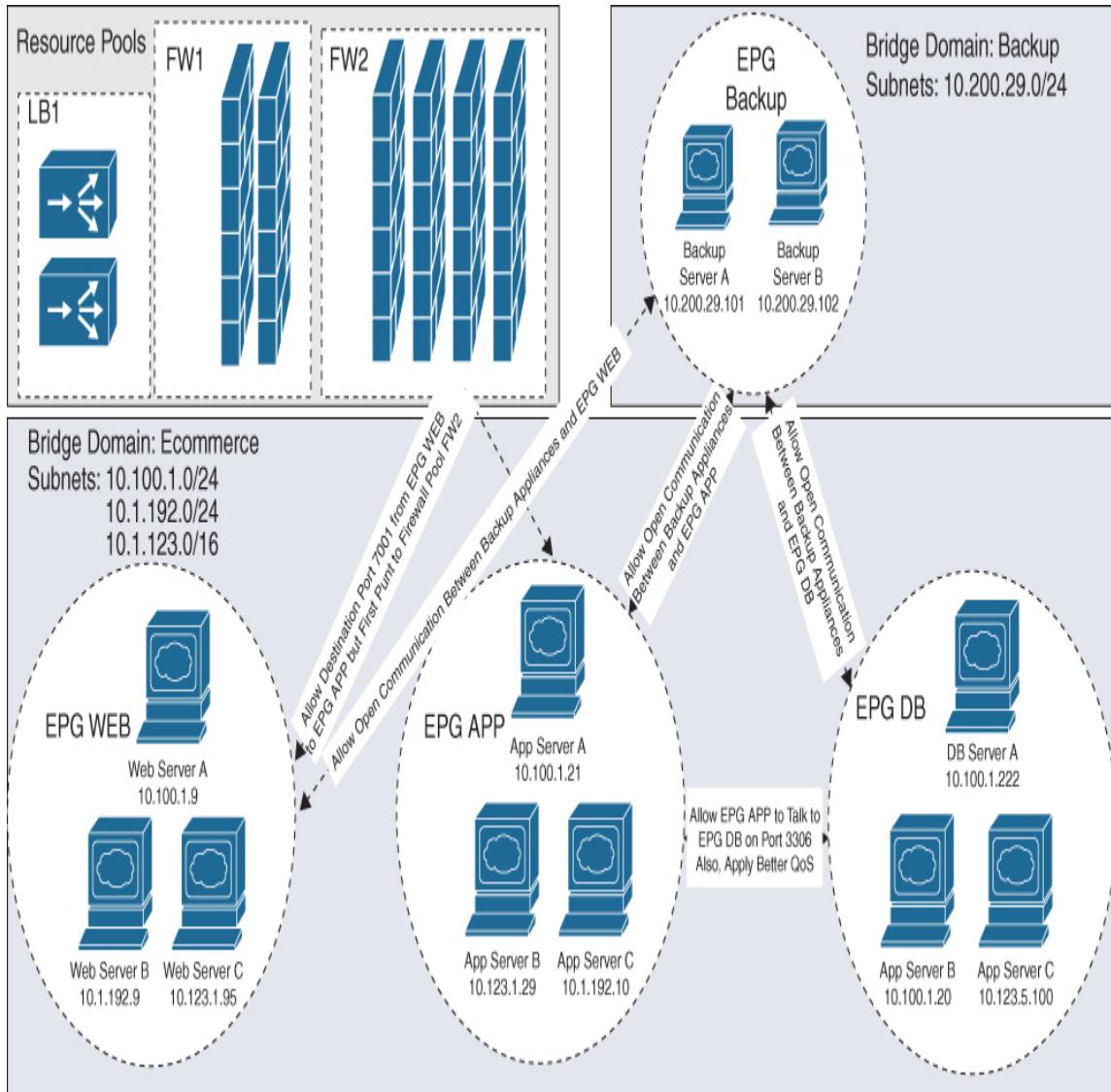


Figure 5-8 Selective Policy Application at the EPG Boundary

Note

Endpoints within an EPG can reside in different subnets as long as all the subnets are associated with the same bridge domain to which the EPG is associated.

Configuring Bridge Domains, Application Profiles, and EPGs

Because EPGs need to be associated with bridge domains and application profiles need to be created before EPGs, the ideal order of operation is to first create bridge domains, then application profiles, and finally EPGs.

[Figure 5-9](#) shows how to navigate to the Create Bridge Domain wizard. Within the Tenants view, open Networking, right-click Bridge Domains, and select Create Bridge Domain.

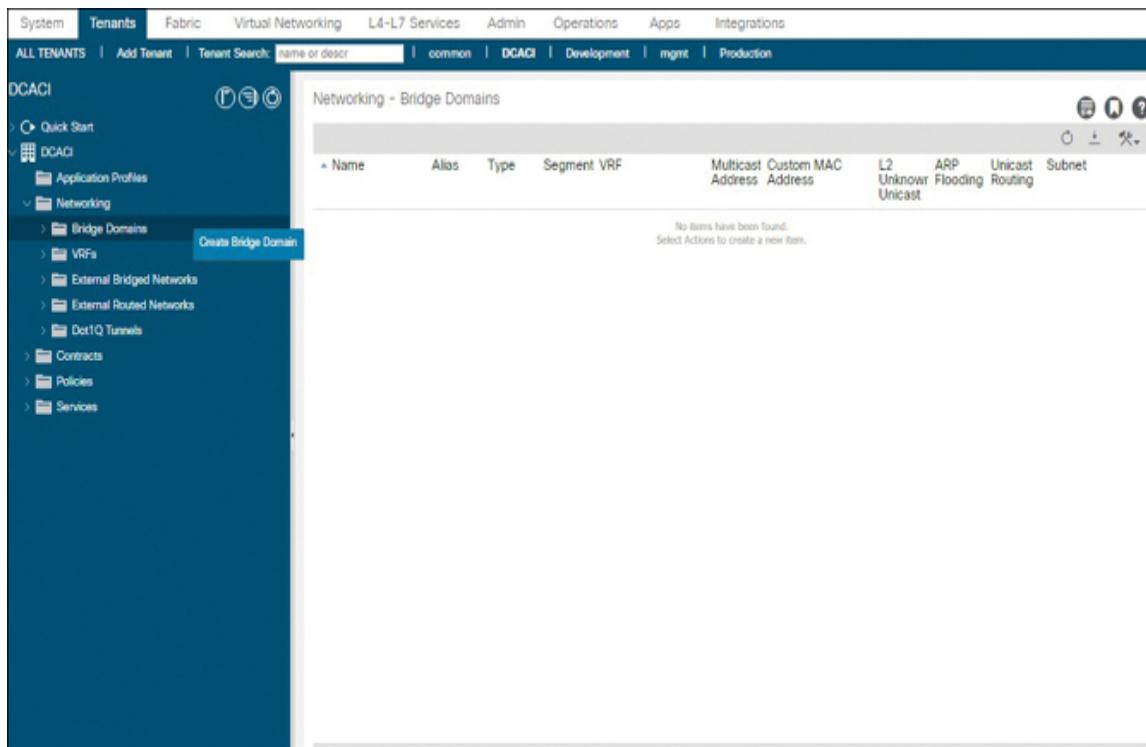


Figure 5-9 Navigating to the Create Bridge Domain Wizard

In the first page of the Create Bridge Domain wizard, which relates to general aspects of the bridge domain, enter a name for the bridge domain and associate the bridge

domain to a VRF instance, as shown in [Figure 5-10](#). Then click Next.

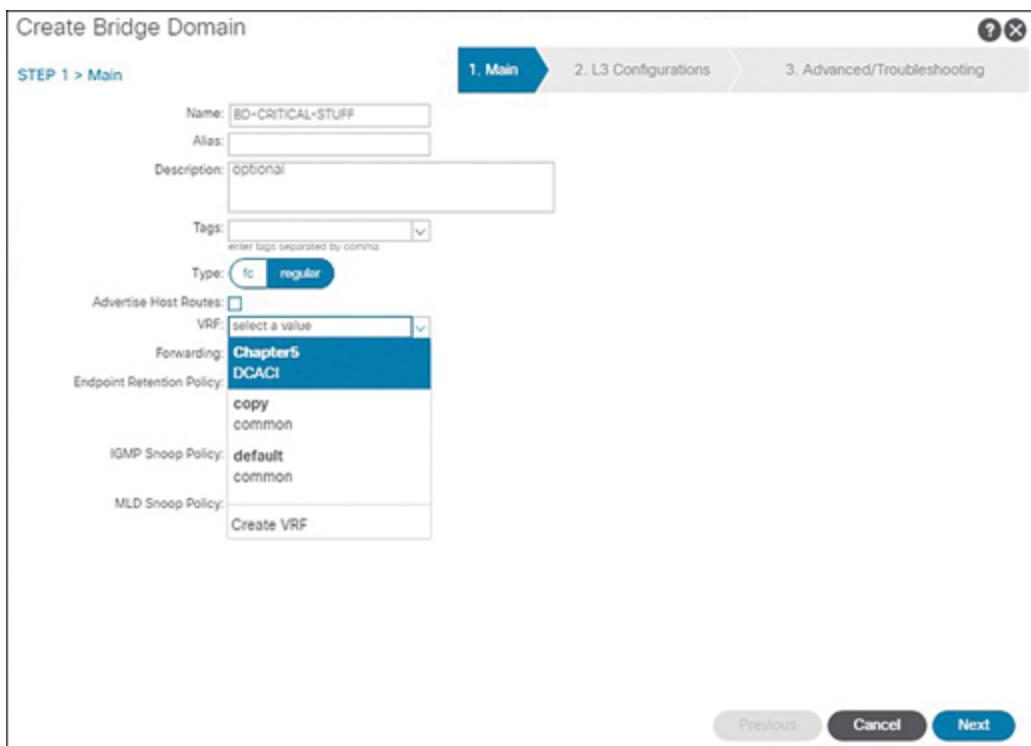


Figure 5-10 *Create Bridge Domain Wizard, Page 1*

[Figure 5-11](#) shows the second page of the Create Bridge Domain wizard, where you enter Layer 3 configurations for the bridge domain. Click the + sign in the Subnets section to open the Create Subnet page.

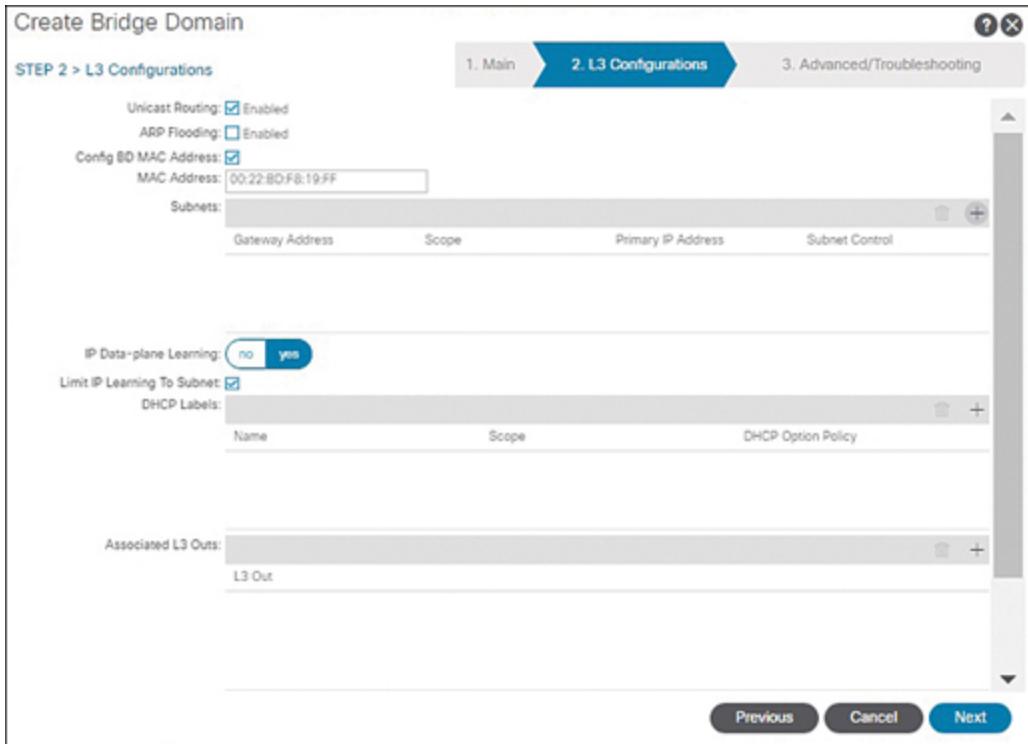


Figure 5-11 Create Bridge Domain Wizard, Page 2

In the Create Subnet page, enter the default gateway IP address of the desired subnet, using CIDR notion (see [Figure 5-12](#)). This gateway IP address will be created in ACI when certain conditions are met. Click OK to return to page 2 of the Create Bridge Domain wizard. Then click Next to move to the last page of the Create Bridge Domain wizard. Note that you can assign multiple subnet IP addresses to each bridge domain.

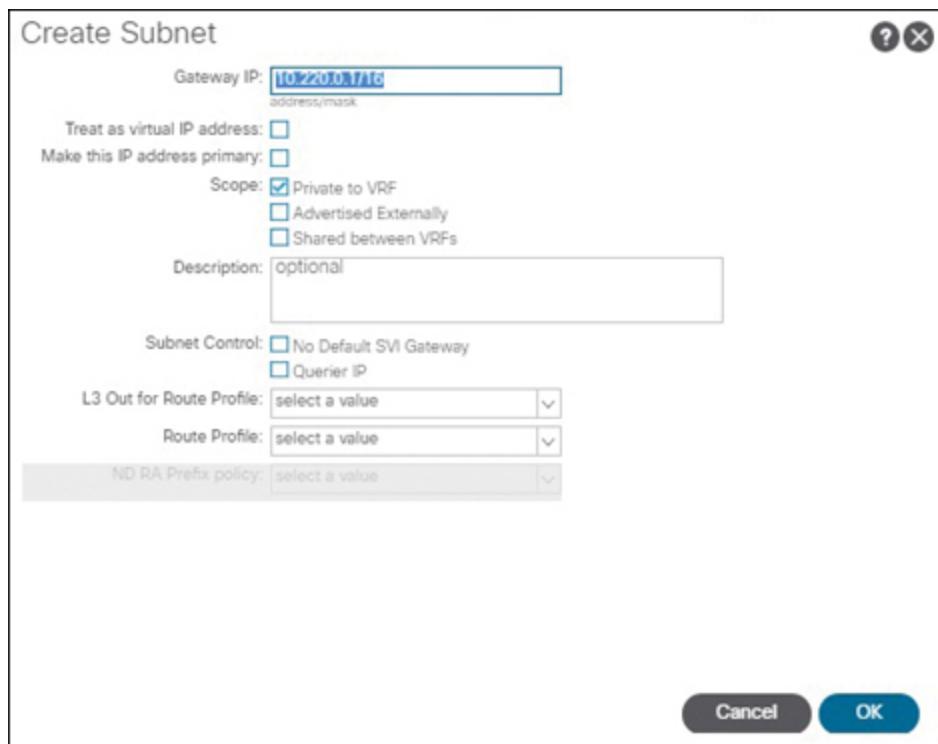


Figure 5-12 Creating a Subnet for the Bridge Domain

Figure 5-13 shows the final page of the Create Bridge Domain wizard, which provides advanced bridge domain settings. Click Finish.

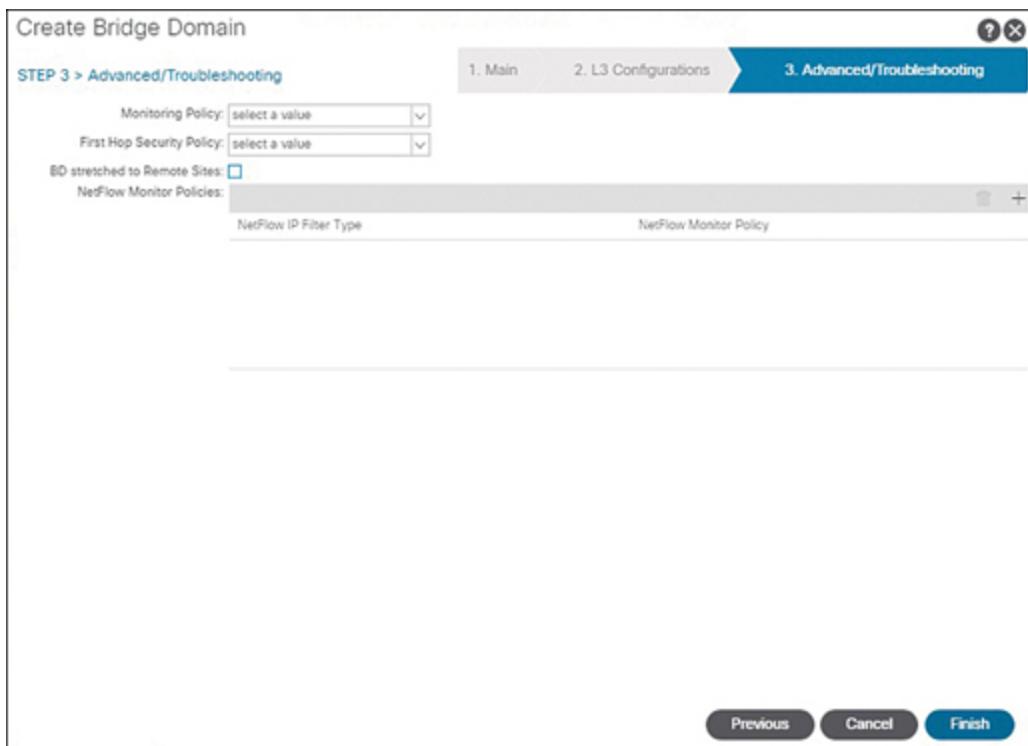


Figure 5-13 *Create Bridge Domain Wizard, Page 3*

After you create bridge domains, you can create application profiles. [Figure 5-14](#) shows how to navigate to a tenant, right-click Application Profile, and select Create Application Profile.

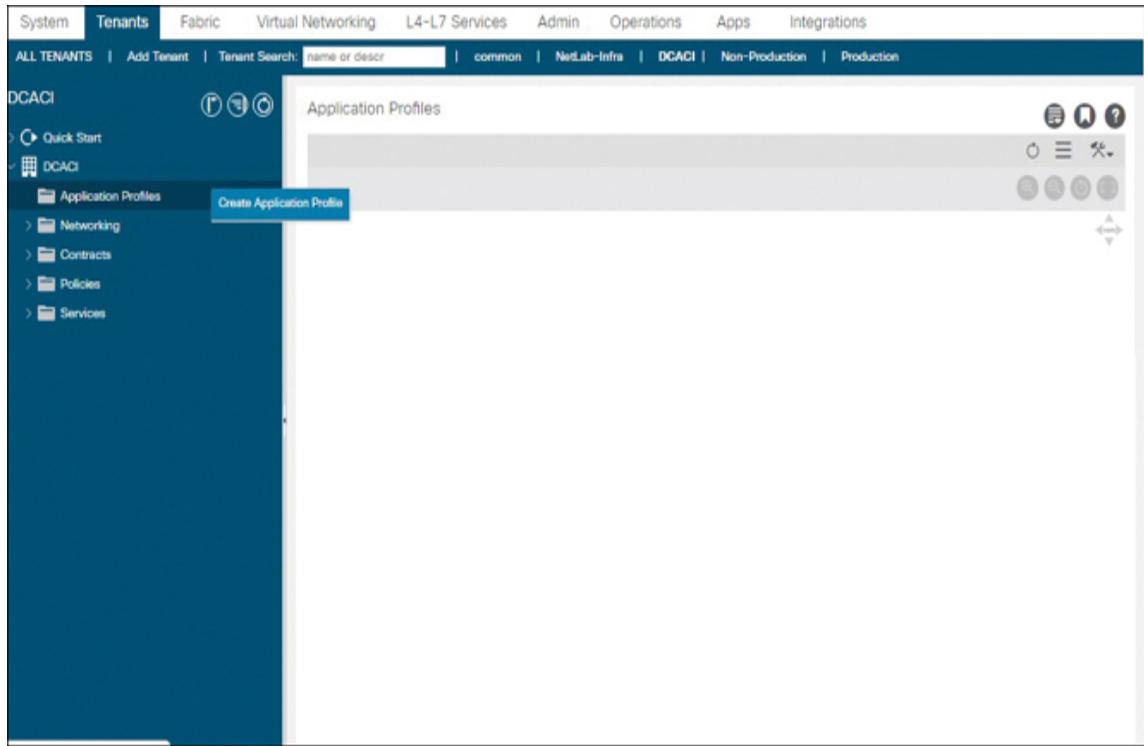


Figure 5-14 Navigating to the Create Application Profile Wizard

In the Create Application Profile wizard, enter a name for the application profile and click Submit, as shown in [Figure 5-15](#).

Create Application Profile

Name:	Critical-Application
Alias:	
Description:	optional
Tags:	enter tags separated by comma
Monitoring Policy:	select a value

EPGs

Name	Alias	BD	Domain	Switching Mode	Static Path	Static Path VLAN	Provided Contract	Consumed Contract

Cancel Submit

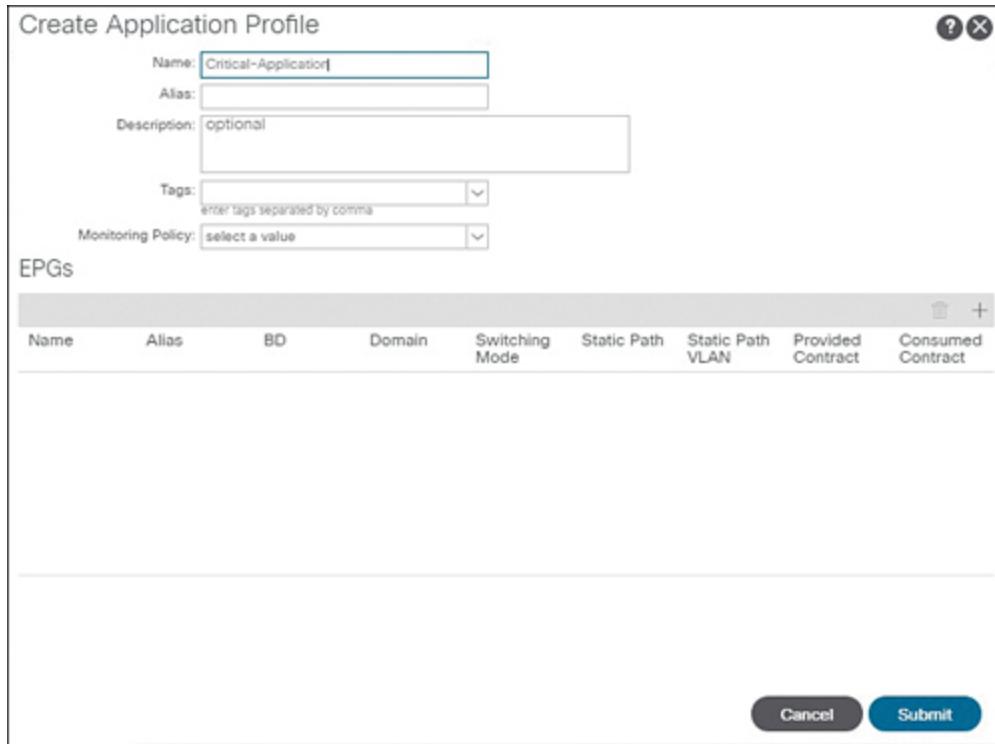


Figure 5-15 Creating an Application Profile

Once an application profile has been created, you can create EPGs within the application profile. Navigate to the Tenants view, right-click the desired application profile under which EPGs should be created, and select Create Application EPG to access the Create Application EPG wizard. As shown in [Figure 5-16](#), you enter a name for an EPG and click Finish.

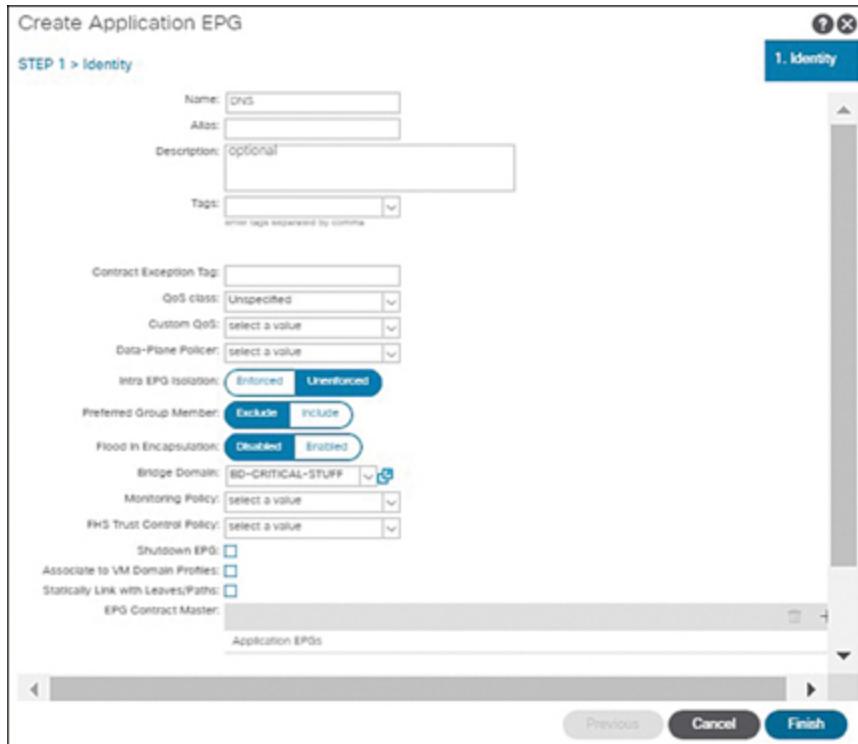


Figure 5-16 Create Application EPG Wizard

Classifying Endpoints into EPGs

In the backend, EPGs and bridge domains each correlate to VXLAN IDs, which are not supported on most server operating systems. ACI needs a mechanism to classify or place endpoint traffic it receives on switch ports into the proper EPGs. ACI most often classifies endpoints and associated traffic into EPGs through the encapsulations that have been mapped to the leaf interfaces on which traffic arrives.



VLAN IDs and VXLAN IDs are forms of encapsulation that ACI uses to classify Ethernet traffic into EPGs.

Note

A uSeg EPG is a specific type of endpoint group that uses endpoint attributes as opposed to encapsulations to classify endpoints. For instance, if you wanted to dynamically classify all virtual machines that run a specific operating system (OS) into an EPG, you could use uEPGs. Classifying endpoints into uSeg EPGs is particularly useful when there is a need to leverage endpoint attributes defined outside ACI (for example, VMware vCenter) to whitelist communication. Use of uSeg EPGs for whitelisting is called *microsegmentation*.

[Figure 5-17](#) provides a simple but realistic depiction of a tenant, a VRF instance, bridge domains, EPGs, and subnets, and it shows how VLAN encapsulations are used to classify endpoints into a given EPG. The encapsulation configurations shown are commonly configured within tenants and at the EPG level. In this case, the tenant administrator has decided to map the EPG named DNS to port channel 1 on Leaf 101 using the VLAN 101 encapsulation. Likewise, the same encapsulation has been mapped to port 1 (Eth1/1) on Leaf 102 to classify server traffic in VLAN 101 into the DNS EPG. Encapsulations can also be mapped to virtual port channels.

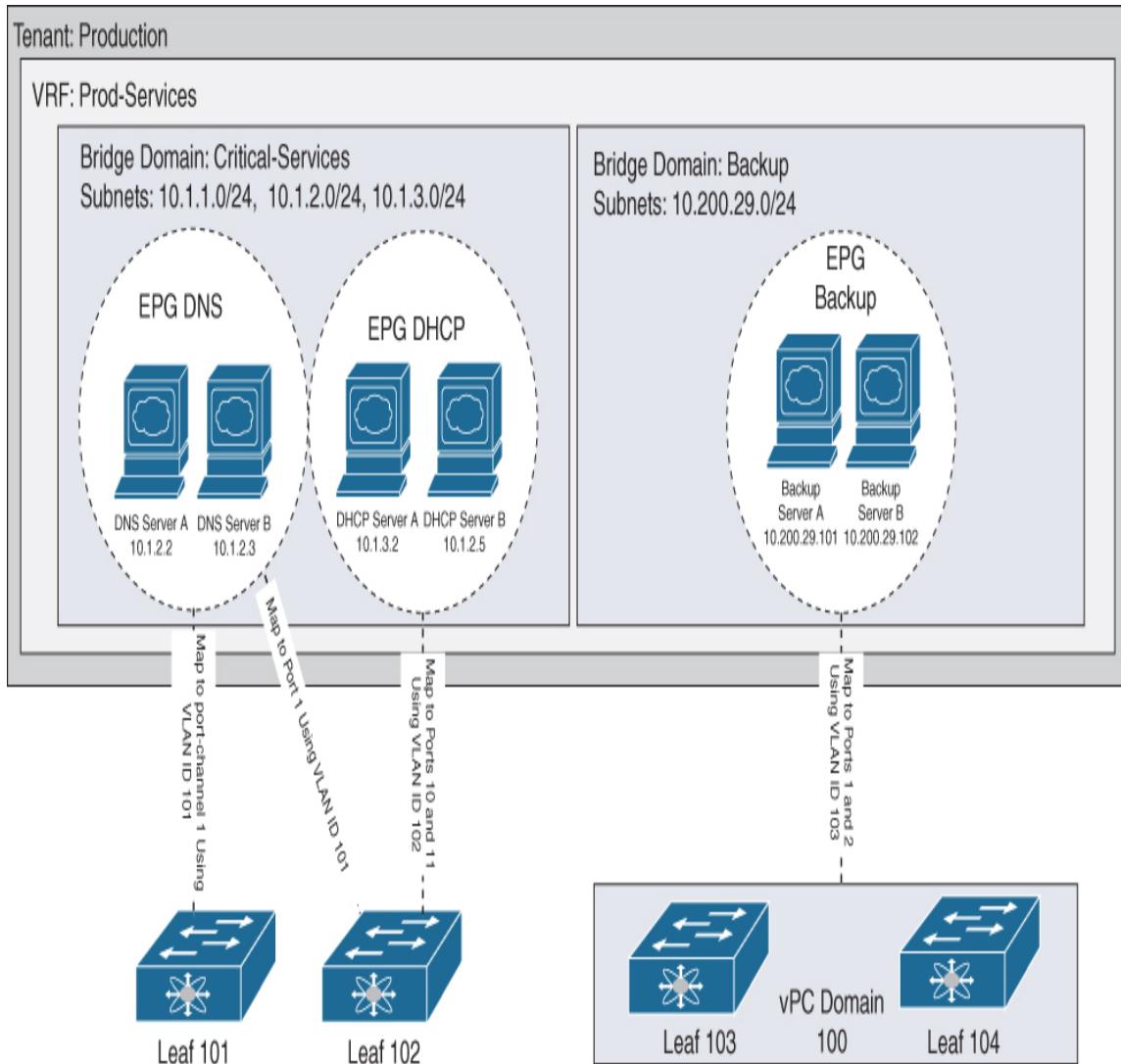


Figure 5-17 Mapping EPGs to Encapsulations

There is no hard requirement for an EPG to be mapped to a single encapsulation across all leaf switches within an ACI fabric. However, it is common practice, and most companies do it to promote standardization.

Note

If you do a cursory review of the ACI GUI, you might wonder why there is a subnet definition section under the EPG view if bridge domains are the construct that

defines subnets. Although you can define a subnet under an EPG (and definition of subnets under EPGs is required in some cases), it is still the switch virtual interface (SVI) associated with the bridge domain and not the EPG that handles routing. Later chapters expand on this simplistic explanation.

APIC CLI Configuration of Tenant Objects

The GUI-based configurations performed in this section can be completed using the CLI commands depicted in [Example 5-2](#).

As a review, the command **tenant DCACI** in [Example 5-2](#) creates a tenant named DCACI. Under the tenant, the ACI engineer creates a VRF instance called Chapter5 by using the **vrf context Chapter5** command. The **exit** command that follows is required because a bridge domain is not a VRF subtree but a tenant child object. The command **bridge-domain BD-CRITICAL-STUFF** creates a bridge domain named BD-CRITICAL-STUFF under the tenant, and **vrf member Chapter5** associates the bridge domain with the Chapter5 VRF. The command **interface bridge-domain BD-CRITICAL-STUFF** is used to signal the intent to create one or more SVIs under the bridge domain BD-CRITICAL-STUFF. The **ip address** subcommand creates an SVI with the address 10.220.0.1/16 as the default gateway. Although the subnet has been created as a secondary subnet, it could as well have been defined as the primary IP address with the **secondary** keyword omitted from the command.

Example 5-2 *CLI Equivalents for Configurations Performed in This Section*

[Click here to view code image](#)

```
apic1# show run tenant DCACI
# Command: show running-config tenant DCACI
# Time: Sat Sep 21 21:12:14 2019
  tenant DCACI
    vrf context Chapter5
      exit
    bridge-domain BD-CRITICAL-STUFF
      vrf member Chapter5
      exit
    application Critical-Application
      exit
  interface bridge-domain BD-CRITICAL-STUFF
    ip address 10.220.0.1/16 secondary
    exit
  exit
```

Contract Security Enforcement Basics

ACI performs whitelisting out of the box. This means that, by default, ACI acts as a firewall and drops all communication between EPGs unless security rules (most commonly contracts) are put in place to allow communication.

ACI security policy enforcement generally involves the implementation of contracts, subjects, and filters.

Contracts, Subjects, and Filters

In the ACI whitelisting model, all inter-EPG communication is blocked by default unless explicitly permitted. Contracts, subjects, and filters complement each other to specify the

level of communication allowed to take place between EPGs. These constructs can be described as follows:



- **Filter:** The job of a *filter* is to match interesting traffic flows. The EtherType, the Layer 3 protocol type, and Layer 4 ports involved in communication flows can all be used to match interesting traffic using *filter entries*. Filters can be defined to be relatively generic to enable extensive reuse.
- **Subject:** Once filters are defined, they are linked to one or more *subjects*. A subject determines the actions that are taken on the interesting traffic. Should matching traffic be forwarded, dropped, or punted to a firewall or load balancer? Should the traffic that has been matched by filters be reclassified into a different QoS bucket? These can all be defined by subjects. A subject can also define whether corresponding ports for return traffic should be opened up.
- **Contract:** A *contract* references one or more subjects and is associated directionally to EPGs to determine which traffic flows are bound by the contract. Contracts are scope limited and can also be configured to modify traffic QoS markings.

Because the concept of ACI contracts can be difficult to grasp, some examples are in order. [Figure 5-18](#) shows an example of how you might set up filters, subjects, and contracts to lock down a basic multitier application. Applications also require connectivity to critical services such as DNS and some method to enable connectivity for outside users. This figure does not show contracts beyond

those needed for the various tiers of the application to communicate.

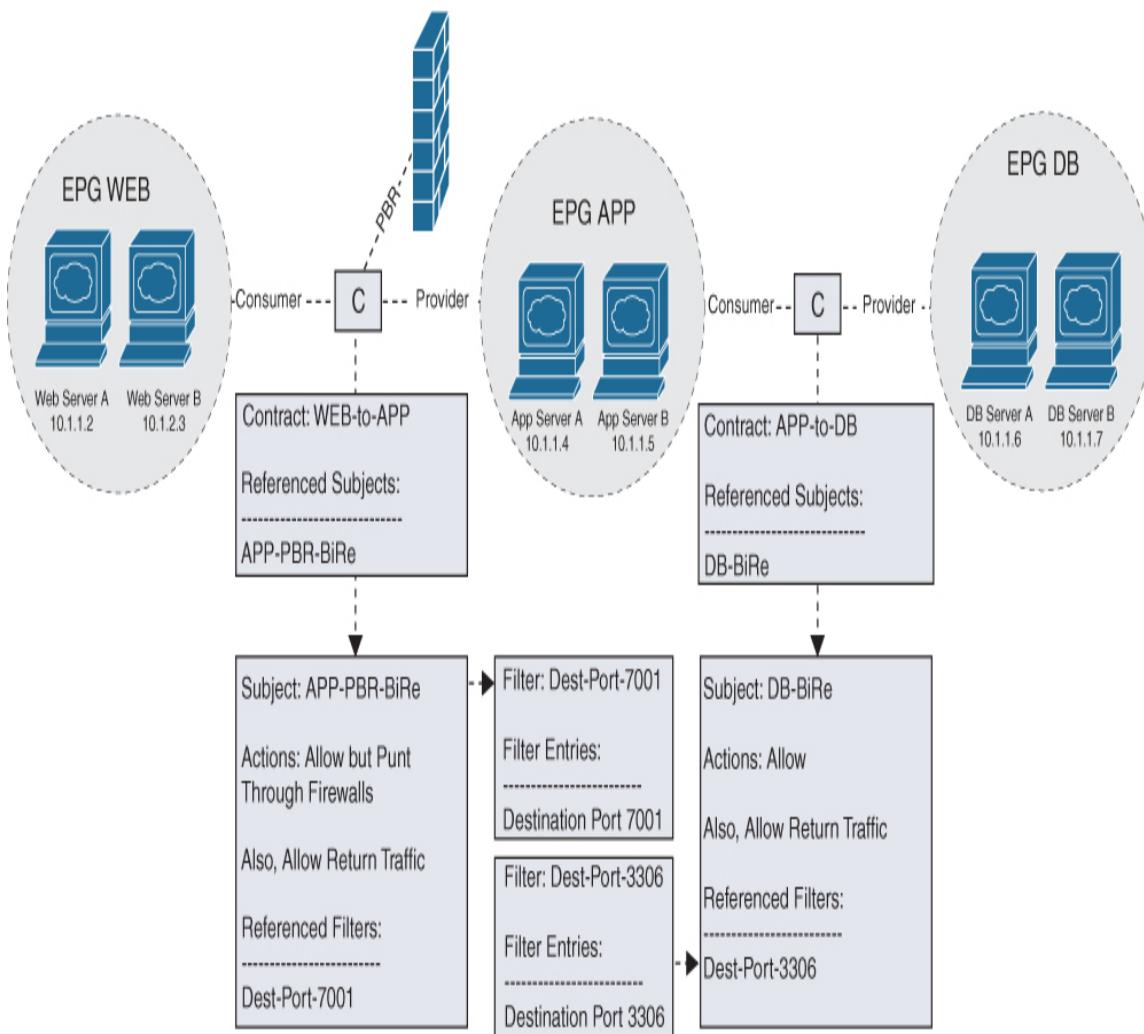


Figure 5-18 *Filters, Subjects, and Contracts*

For now, do not worry about implementation procedures for contracts, subjects, and filters. Implementation of these objects is covered in [Chapter 8, “Implementing Tenant Policies.”](#)



ACI allows open communication between endpoints residing in a single EPG (intra-EPG) by default without the need for contracts, but intra-EPG communication can also be locked down. [Figure 5-16](#), presented earlier in this chapter, shows the Intra EPG Isolation configuration option, which is set to Unenforced by default. There are very compelling use cases for setting Intra EPG Isolation to Enforced. For example, management stations that would reside either outside an ACI fabric or in a separate EPG may need to communicate with server CIMC out-of-band connections, but CIMC ports across multiple servers have no need to cross-communicate. Where there is no need for endpoints within an EPG to communicate with one another, intra-EPG isolation can be implemented on the given EPG. This feature uses private VLAN functionality without the headache of administrators having to define primary and secondary VLANs for bare-metal connections.

Contract Direction



ACI contracts are directional because TCP/IP communication is inherently directional. A client service initiates communication with a server. The server is a **provider** of a service to the client machine, and the client is a **consumer** of a service. (A sample directional application of contracts is presented in [Figure 5-18](#).)

All communication within data centers conforms to this provider/consumer model. Although a web server provides services to users, it is also consuming services itself. For example, it may attempt to initiate communication with a backend database server, NTP servers, and DNS servers. In

these cases, the web server acts as a client machine. Any contracts that allow outside users to access web services on the web server should be applied to the web server EPG in the provider direction. However, any contracts that allow the web server to communicate with other servers for NTP, DNS, and backend database access need to be applied to the web EPG in the consumer direction and to the database server, NTP servers, and DNS servers in the provider direction.

Contract Scope

A **contract scope** is a condition that determines whether a contract can be enforced between EPGs. Options for contract scope are as follows:



- **Application profile:** A contract with an application profile scope can be enforced between EPGs if they reside within the same application profile.
- **VRF:** A contract with a VRF scope can be enforced between EPGs if they reside within the same VRF instance. EPGs can be in different application profiles.
- **Tenant:** A contract with a tenant scope can be applied between EPGs if they are all in the same tenant. The EPGs can be in different VRFs and application profiles.
- **Global:** A contract with a global scope can be applied between any EPGs within a fabric and can be exported between tenants to enable cross-tenant communication. If a global scope contract is placed in the common tenant, it can enable cross-tenant

communication without the need to be exported and imported between tenants.

To better understand contract scopes, reexamine [Figure 5-18](#). Notice that Web Server A and Web Server B can communicate with one another without contracts because they are in the same EPG. An administrator who wanted to prevent all communication between endpoints within an EPG could block all intra-EPG traffic at the EPG level. However, this is not always desirable. Sometimes, a subset of endpoints within an EPG might be in a clustered setup and need to communicate, while others should not be allowed to communicate. Moreover, the contracts shown in [Figure 5-18](#) enable open communication between Web Server A and App Server B, with the hope that the firewall blocks such communication if it is not desired. If the suffixes A and B denote different applications, the contracts depicted would be considered suboptimal because ACI would allow communication across different applications.

As an alternative, consider [Figure 5-19](#). All endpoints suffixed with the letter A form a LAMP stack and have been placed into an application profile called LAMP1. Similarly, endpoints suffixed with the letter B form a separate three-tier application and have been placed into LAMP2. Moreover, the scope of the contracts, which was unclear from the previous example, has been clarified to be Application Profile. With this modification, even if a slight configuration mistake were to occur in contract configuration and its application to the EPGs (for example, if all ports were erroneously opened), the mistake would be scope limited to each application profile. In other words, various tiers of different application profiles would still be unable to communicate. Therefore, you can translate the logic applied by the contract scope to mean “apply this

contract between EPGs only if they are all in the same application profile.”

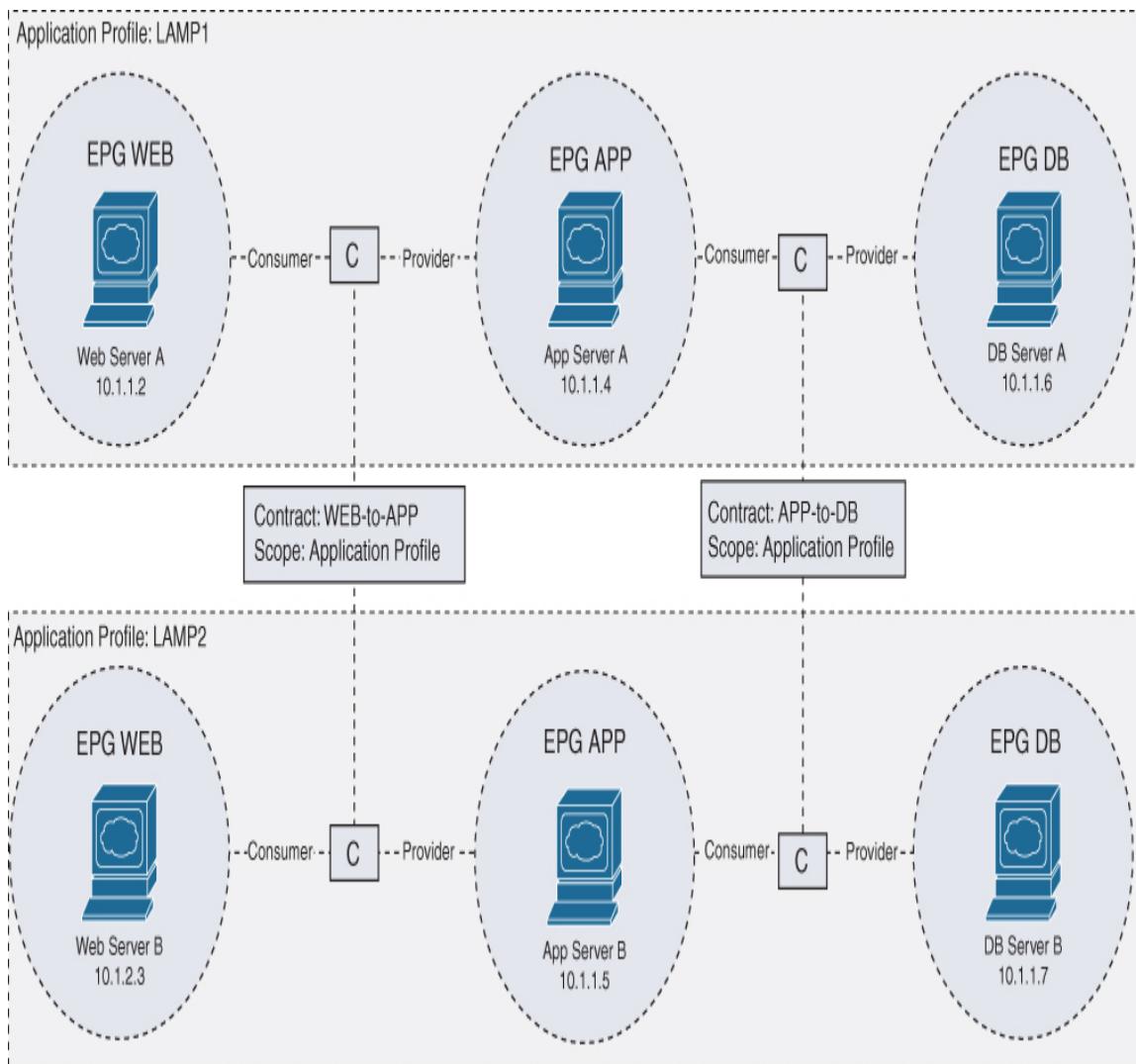


Figure 5-19 Contract Scope Example

As an extension of this example, what scope would you need to use in a new contract applied to all of the depicted EPGs, assuming that the contract seeks to allow them to communicate with an NTP server that is in a separate VRF instance within the same tenant? If you answered that the scope needs to be Tenant, you would be right. What scope would have to be defined if the NTP server were in a different tenant? The answer in that case would be Global.

Zero-Trust Using EPGs and Contracts

A zero-trust network architecture is an information security framework originally proposed by research and advisory firm Forrester that addresses the inherent weakness of a perimeter-focused approach to security by assuming no default trust between entities.

Attainment of a zero-trust data center is the primary security objective of EPGs and contracts. As noted earlier, ACI assumes no trust by default between EPGs unless the desired communication has been whitelisted.

Objects Enabling Connectivity Outside the Fabric

Whereas bridge domains, EPGs, and other constructs introduced in this chapter enable the deployment of applications and communication of application tiers, at some point, tenant endpoints need to communicate with the outside world. External EPGs and L3Out objects play a key role in enabling such communication.

External EPGs



An **external EPG** is a special type of EPG that represents endpoints outside an ACI fabric, such as user laptops, campus IoT devices, or Internet users. There are many reasons you might want to classify traffic outside ACI. One reason to do so is to be able to apply different security policies to different sets of users. External EPGs classify

outside traffic using subnets, but the subnets can be as granular and numerous as needed.

[Figure 5-20](#) shows three external EPGs that are allowed different levels of access to servers within an ACI fabric. Any traffic sourced from IP addresses defined in the external EPG named EXT-ADMINS will be allowed access to all the depicted servers via SSH, HTTPS, and RDP, but all other internal users classified into the external EPG called EXT-INTERNAL will be limited to HTTPS access to the web server. All users sourcing traffic from the Internet will be classified into the external EPG called EXT-INTERNET and will therefore be denied any form of access to these specific servers because no contracts permitting communication have been associated between the servers and EXT-INTERNET.

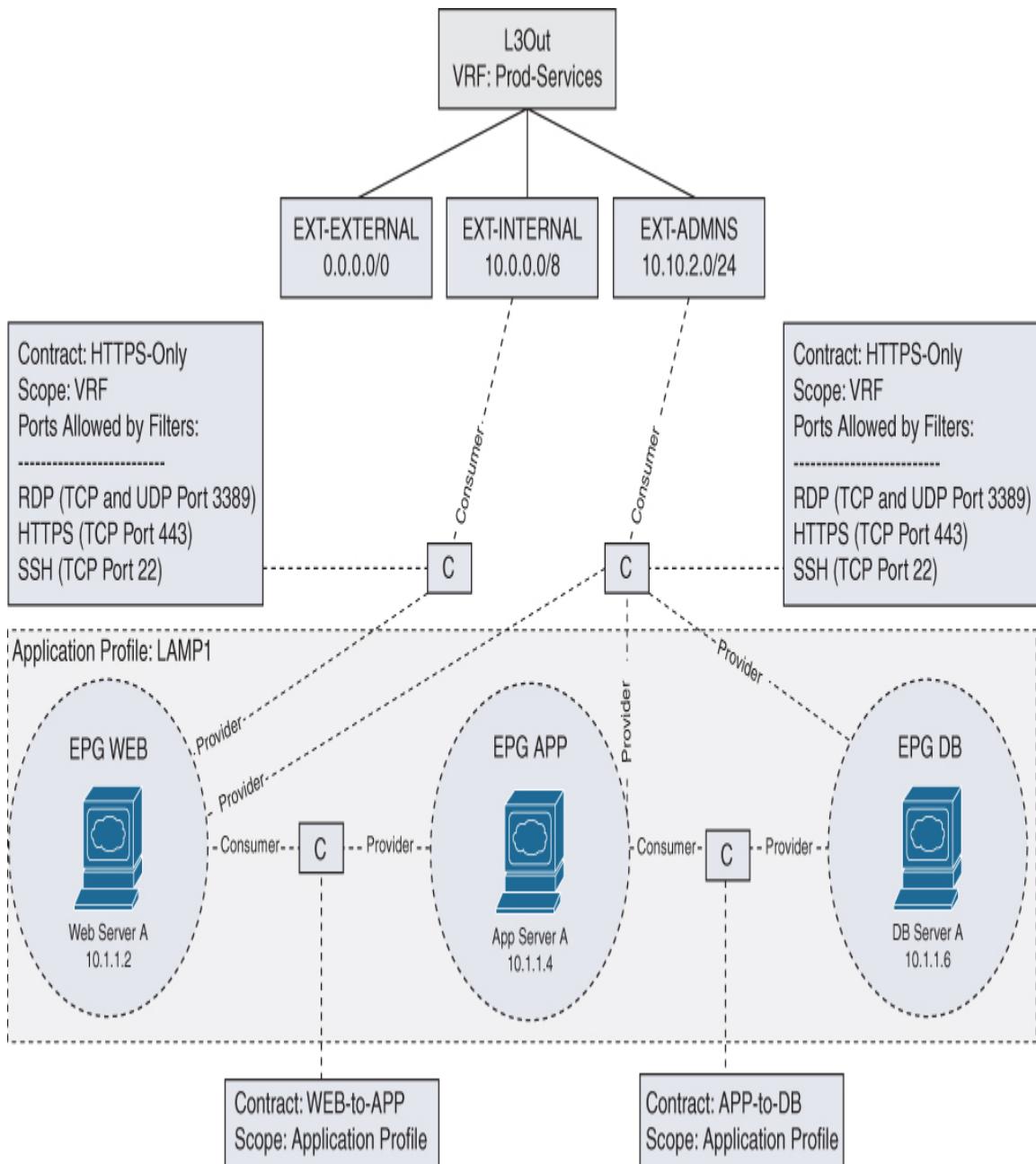


Figure 5-20 Controlling Access to ACI Fabrics by Using External EPGs

One point that is important to clarify here is that external EPG subnets are longest prefix-match subnets. Therefore, EXT-INTERNET, which consists of the 0.0.0.0/0 subnet, classifies all endpoints out on the Internet but not internal

subnets in the more specific 10.0.0.0/8 range allocated to EXT-INTERNAL.

Expanding on this concept, it is important to understand that any given outside endpoint will be classified to one and only one external EPG. Just because the administrator group defined by EXT-ADMINS at 10.10.2.0/24 also falls within the 10.0.0.0/8 range does *not* mean that administrators will have some of their access removed to reflect the access levels of the 10.0.0.0/8 range. Likewise, if EXT-INTERNAL were allocated more access than EXT-ADMINS, the 10.10.2.0/24 administrator subnet would not inherit expanded access.

So, what happens if an administrator associates a particular subnet with multiple external EPGs? ACI triggers a fault, and the second subnet allocation to an external EPG is invalidated. The only exception to this rule is the 0.0.0.0/0 subnet. Regardless of this exception, deployment of multiple external EPGs that reference the same subnet is bad practice.



External EPGs, sometimes referred to as *outside EPGs*, classify traffic based on a longest-prefix match, and any given outside endpoint will be classified into the most specific applicable external EPG that has been defined.

Note

Other types of external EPGs exist. The type of external EPG used for classification of external traffic that is described here is configured with the Scope value External Subnets for the External EPG. [Chapter 9](#),

“[L3Outs](#),” addresses the other Scope settings that are available.

Also, as shown in [Figure 5-20](#), external EPGs associate with objects called Layer 3 Outs, which in turn bind to VRF instances.

Layer 3 Outside (L3Out)



An **L3Out** is an object that defines a route peering or a series of route peerings to allow route propagation between ACI and an external Layer 3 switch, router, or appliance. BGP, OSPF, and EIGRP are all supported protocols for use on L3Outs. Static routes pointing outside ACI can also be configured on L3Outs.

A regular L3Out is configured within a tenant and is bound to a single VRF instance. A number of specialized L3Outs can be created in the infra tenant, which can advertise routes from multiple ACI VRF instances to the outside world. This book focuses on regular L3Outs.

Tenant Hierarchy Review

[Figure 5-21](#) provides an overview of the tenant hierarchy and the relationship between the objects outlined so far in this chapter. Each relationship between tenant objects is shown to be either a $1:n$ (one-to-many) relationship or an $n:n$ (many-to-many) relationship. [Figure 5-21](#) shows, for example, that any one bridge domain can be associated with one and only one VRF instance. However, any one bridge domain can also have many subnets associated with

it, so a bridge domain can have a $1:n$ relationship with subnets.

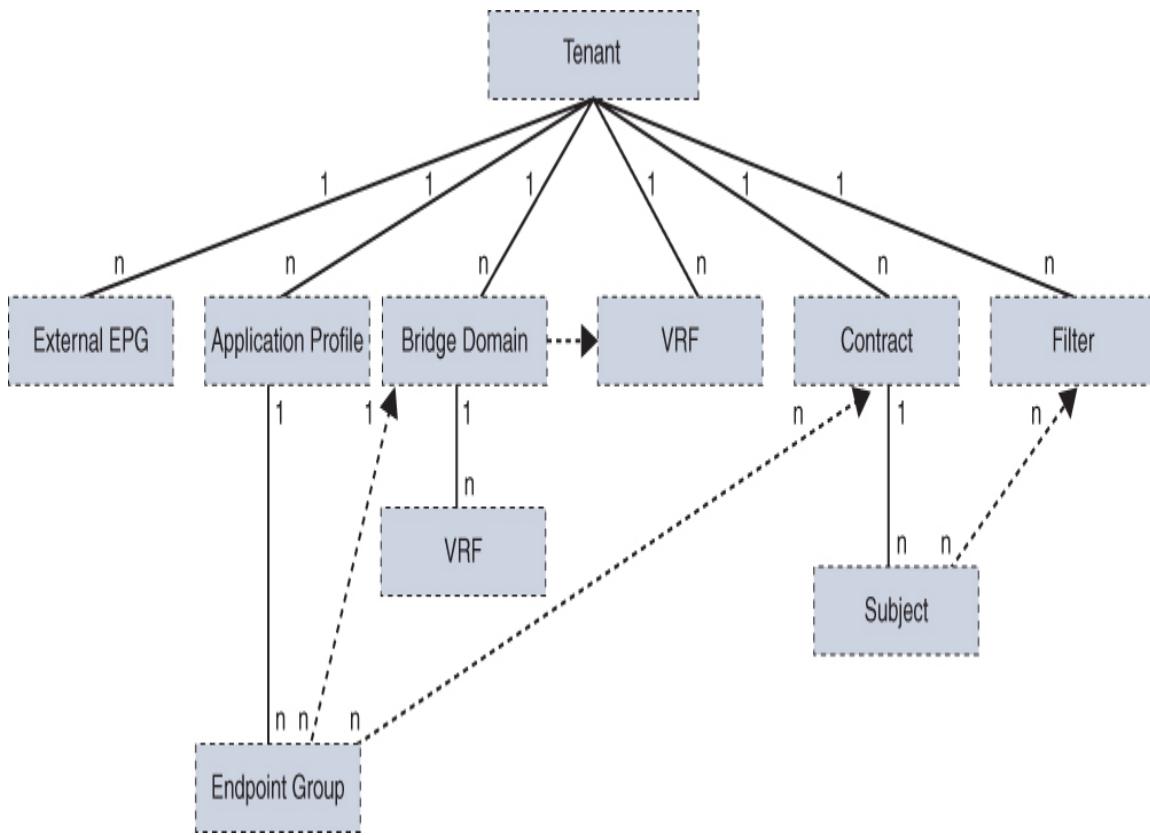


Figure 5-21 Objects That Reside in Tenants and Their Relationships

Exam Preparation Tasks

As mentioned in the section “How to Use This Book” in the Introduction, you have a couple of choices for exam preparation: [Chapter 17](#), “Final Preparation,” and the exam simulation questions in the Pearson Test Prep Software Online.

Review All Key Topics

Review the most important topics in this chapter, noted with the Key Topic icon in the outer margin of the page. [Table 5-2](#) lists these key topics and the page number on which each is found.



Table 5-2 Key Topics for [Chapter 5](#)

Key Topic Element	Description	Page Number
Paragraph	Defines tenants	133
List	Describes predefined tenants in ACI	134
Paragraph	Defines VRF instances	135
Paragraph	Defines bridge domains	137
Paragraph	Defines EPGs	137
Paragraph	Defines application profiles	138

Paragraph	Describes associations between application profiles, bridge domains, and EPGs	139
Paragraph	Describes practical bridge domain functions	141
Paragraph	Describes practical EPG functions	141
Paragraph	Lists encapsulations in ACI for Ethernet traffic	146
List	Defines contracts, subjects, filters, and filter entries	148
Paragraph	Defines and provides a sample use case for intra-EPG isolation	149
Paragraph	Defines contract direction options	149
List	Lists and describes contract scopes	150
Paragraph	Defines external EPGs	151

Paragraph	Describes method of endpoint classification by external EPGs	153
Paragraph	Defines L3Outs	153

Complete Tables and Lists from Memory

There are no memory tables or lists in this chapter.

Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

tenant

fabric policy

fabric port

access policy

Virtual Machine Manager (VMM) domain

virtual routing and forwarding (VRF) instance

application profile

bridge domain (BD)

endpoint group (EPG)

filter

subject

contract

contract scope

consumer

provider
external EPG
L3Out

Chapter 6

Access Policies

This chapter covers the following topics:

Pools, Domains, and AAEPs: This section outlines the significance of multitenancy-centric objects in the Access Policies view.

Policies and Policy Groups: This section addresses the grouping of interface policies and switch policies into reusable policy groups.

Profiles and Selectors: This section explains the role of profiles and selector objects in configuring ports and enabling stateless networking.

Bringing It All Together: This section summarizes how the critical objects detailed in this chapter link tenancy to the underlying infrastructure.

This chapter covers the following exam topics:

- 1.5 Implement ACI policies
 - 1.5.a access

Aside from tenant objects, the most important objects ACI administrators deal with on a regular basis are those that relate to access policies.

The objects detailed in this chapter are critical to the configuration of switch ports. They enable service providers and central IT to control the encapsulations, the types of external devices, and the switch ports to which tenant administrators are allowed to deploy endpoints.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 6-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Questions.”](#)

Table 6-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Pools, Domains, and AAEPs	1-5
Policies and Policy Groups	6-8
Profiles and Selectors	9, 10

Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

- 1.** Which of the following objects is used when attaching a bare-metal server to an ACI fabric?
 - a.** External bridge domain
 - b.** VMM domain
 - c.** Routed domain
 - d.** Physical domain
- 2.** True or false: When an administrator assigns a VLAN pool to a domain that is associated with an AAEP and the administrator then assigns the AAEP to switch interfaces, the VLANs in the VLAN pool become trunked on all the specified ports.
 - a.** True
 - b.** False
- 3.** True or false: A VMM domain allows dynamic binding of EPGs into virtualized infrastructure.
 - a.** True
 - b.** False
- 4.** Before a tenant administrator maps an EPG to ports and encapsulations, he or she should first bind the EPG to one

or more _____.

- a.** endpoints
- b.** VRF instances
- c.** AAEPs
- d.** domains

5. Which of the following statements is correct?

- a.** An EPG cannot be assigned to more than one domain.
- b.** An EPG can be bound to multiple domains, but the domains ideally should not reference overlapping VLAN pools.
- c.** An EPG cannot have static mappings to physical ports.
- d.** An EPG can be directly associated with a VRF instance.

6. Which of the following protocols does ACI use for loop prevention?

- a.** Spanning Tree Protocol
- b.** LACP
- c.** MCP
- d.** DWDM

7. A port channel interface policy group configuration has been assigned to a switch. An engineer has been tasked with creating a second port channel with equivalent configurations on the same switch. He decides to reuse the interface policy group and make a new port assignment using a new access selector name. Which of the following statements is accurate?

- a.** ACI creates a new port channel because a new access selector is being used.

- b. ACI adds the ports assigned to the new access selector to the previously created port channel bundle.
 - c. ACI triggers a fault and does not deploy the configuration.
 - d. ACI does not trigger a fault or deploy the configuration.
- 8. True or false: Access (non-aggregated) interface policy groups are fully reusable.
 - a. True
 - b. False
- 9. True or false: Multiple interface profiles can be assigned to a switch.
 - a. True
 - b. False
- 10. Which of the following need to be directly associated with node IDs?
 - a. Interface profiles
 - b. AAEPs
 - c. Switch profiles
 - d. Interface policies

Foundation Topics

Pools, Domains, and AAEPs

While tenant network policies are configured separately from access policies, tenant policies are *not* activated unless their underlying access policies are in place. Therefore, tenants depend on access policies.

Access policies govern the configuration of any non-fabric (access) ports. The term *access policies* in the context of ACI, therefore, should not be understood as the access versus trunking state of a port. In fact, the trunking state of ports is usually determined by encapsulation mappings and is often configured within tenants and not in the access policies view.

Regardless of whether a non-fabric port is expected to function as a trunk port or an access port, configuration of parameters such as interface speed and the protocols to be enabled on the interface are still made under the umbrella of access policies.

In true multitenancy environments with tight role delineation, access policies are configured either by an admin user or a user who has been assigned the access-admin role or a role with equivalent privileges. A user who has been assigned the access-admin role can create the majority of objects in this chapter but would need expanded privileges to create domains.

VLAN Pools

A **VLAN pool** defines the range of VLAN IDs that are acceptable for application to ACI access (non-fabric) ports for a particular function or use. Allocation of VLAN IDs can be performed either statically or dynamically.



With a ***static VLAN allocation***, or *static binding*, an administrator statically maps a specific EPG to a VLAN ID on a port, a port channel, a virtual port channel, or all ports on a switch. With ***dynamic VLAN allocation***, ACI automatically picks a VLAN ID out of a range of VLANs and maps it to an EPG.

Static VLAN allocation is required when configuring access or trunk ports connecting to bare-metal servers and appliances. Dynamic allocation is beneficial in deployments that rely on automated service insertion or VMM integration, where ACI is able to automatically push EPGs into virtualized environments to allow virtualization administrators to assign virtual machines directly to EPGs.

Other forms of pools do exist in ACI, such as VXLAN pools. However, VLAN IDs are the most common form of encapsulation used on ports connecting to servers and appliances as well as outside switches and routers.

Note

Cisco Application Virtual Switch (AVS) and the Cisco ACI Virtual Edge (AVE) support both VLAN and VXLAN as acceptable encapsulations for EPG mappings.

To create a VLAN pool, select **Fabric > Access Policies > Pools** and right-click VLANs. Finally, select Create VLAN Pool, as shown in [Figure 6-1](#).

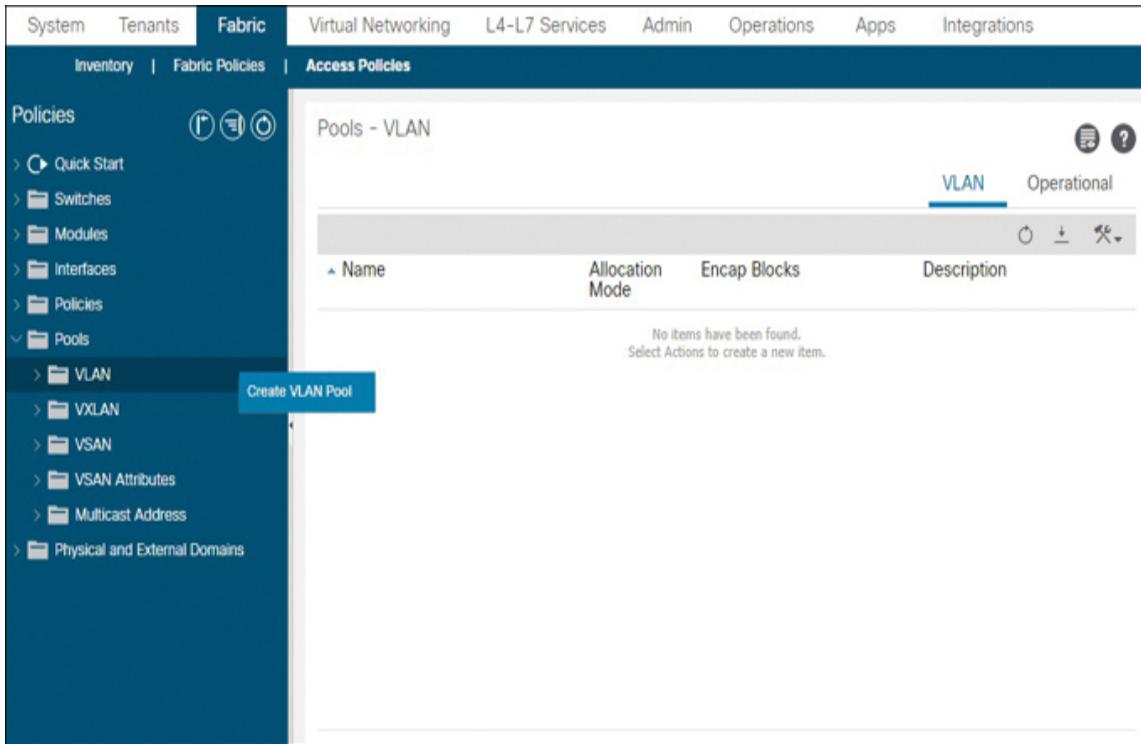


Figure 6-1 Opening the Create VLAN Pool Wizard

In the wizard, type a name for the VLAN pool, select the allocation mode, and then click on the + sign, as shown in [Figure 6-2](#), to open the Create Ranges window.

Create VLAN Pool

Name: DCACI-VLANs

Description: optional

Allocation Mode: Dynamic Allocation Static Allocation

Encap Blocks:

VLAN Range	Allocation Mode	Role

Cancel Submit

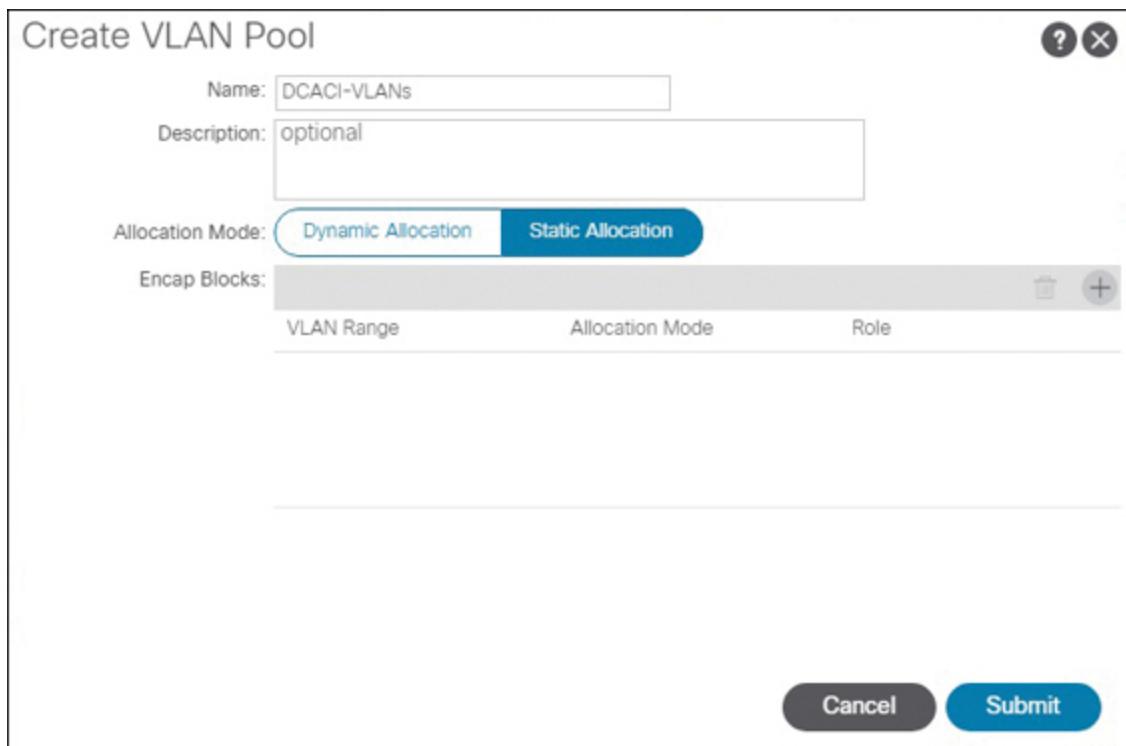


Figure 6-2 Entering the VLAN Pool Name and Allocation Mode

In the Create Ranges window, enter an acceptable range of encapsulations and the role of the range and click OK, as shown in [Figure 6-3](#). A range of VLANs consists of a set of one or more subsequent VLANs. Note that you can create additional ranges and add them to the VLAN pool by navigating to the Create Range window and repeating the process.

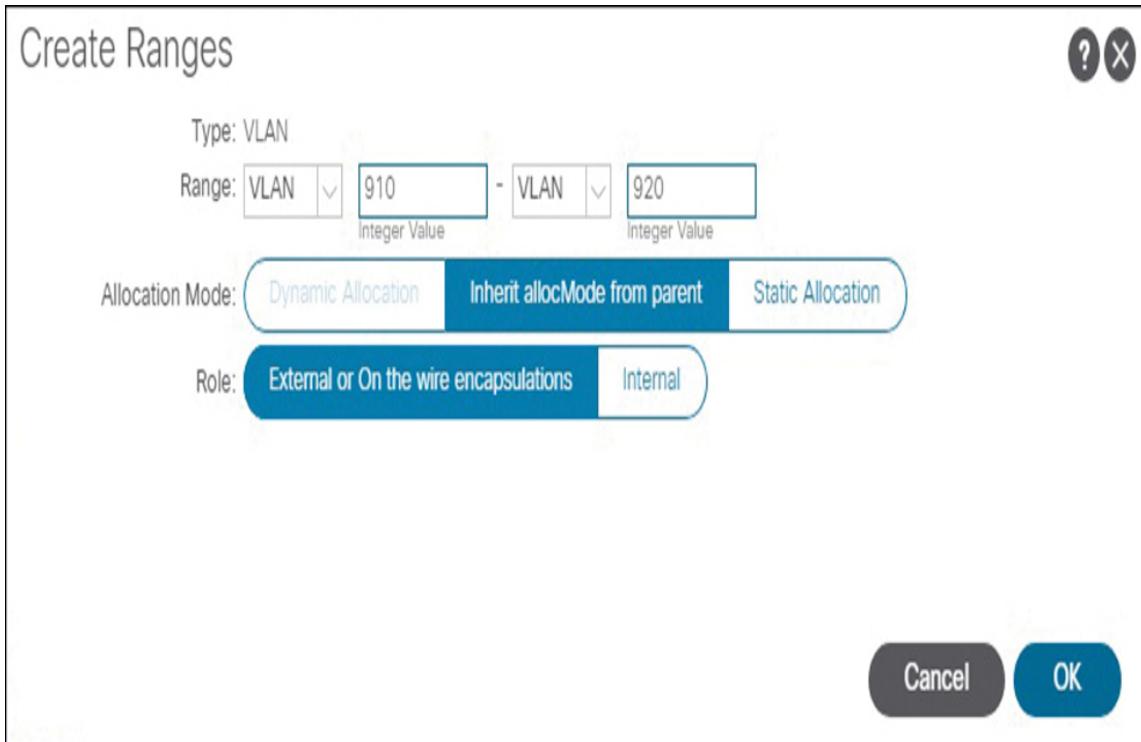


Figure 6-3 Creating a Range of VLANs to Be Added to a VLAN Pool

VLAN ranges created in VLAN pools can take one of two roles:

- **External, or on-the-wire, encapsulation:** Any range of encapsulations used for bare-metal servers or hypervisor uplinks where traffic is trunked outside an ACI fabric is considered to be external or on the wire.
- **Internal:** Private VLAN allocations within very specific virtual switching solutions such as AVE require internal VLAN ranges. Internal VLAN encapsulation ranges reside inside hypervisors and do *not* extend outside hypervisors and onto the wire.

Domains

Key Topic

Domains are the central link between the access policies hierarchy and the tenant hierarchy. A **domain** is the glue that binds tenant EPGs to access and virtual networking policies. With the help of pools, domains determine whether a tenant administrator is even allowed to map an EPG to a certain encapsulation and underlying infrastructure. Each domain points to and consumes a single VLAN pool.

Note that the word *allowed* in the above definition is key. In environments with management multitenancy, an ACI administrator assigns domains to security domains, thereby determining which tenant administrators can bind EPGs to the domain and consume the VLAN IDs defined in the associated VLAN pool. [Chapter 15, “Implementing AAA and RBAC,”](#) covers ACI role-based access control in detail.

The following types of domains within ACI fall within the scope of the Implementing Cisco Application Centric Infrastructure DCACI 300-620 exam:

Key Topic

- **Physical domain:** A **physical domain** governs the attachment of bare-metal servers and appliances that need static VLAN allocations.
- **External bridge domains:** An **external bridge domain** is a type of domain used in attachments to switches outside ACI for Layer 2 connectivity.
- **External routed domains:** An **external routed domain** is a type of domain used in attachments to

switches and routers outside ACI for Layer 3 connectivity.

- **Fibre Channel domains:** This type of domain is used in attachments to servers and storage area networks (SANs) outside ACI for FC or FCoE traffic. In addition to referencing a VLAN pool, a Fibre Channel domain also references a VSAN pool.
- **Virtual Machine Manager (VMM) domain:** A **Virtual Machine Manager (VMM) domain** is a type of domain that enables ACI to deploy EPGs and corresponding encapsulations into virtualized environments.

If you find it difficult to remember the function of domains and why there are numerous types of domains, you can think of the word *how*. The association of domains with objects like bridge domains, AAEPs, and L3Outs tells ACI how a given endpoint is allowed to connect to the fabric.

Keep in mind that a domain, by itself, does not determine whether a tenant administrator can actually map an EPG to an individual server. It just determines the list of VLAN IDs or other forms of encapsulation a tenant administrator has been approved to use for any given type of connectivity (for example, type of domain).

Common Designs for VLAN Pools and Domains

There are many ways to lay out VLAN pools and domains, but three methods are most prevalent in the industry.

For the first type of VLAN pool and domain layout described here, central IT manages everything in ACI. Even though multiple user tenants may have been created, role-based

access control may not be a desired goal of multitenancy in such environments, and all network administrators have full permission to make any changes in ACI. For this reason, a single VLAN pool and domain is created for each device attachment type. [Table 6-2](#) shows an example of this type of design.

Table 6-2 Single VLAN Pool for Each Type of Domain

Domain Name	VLAN Range	Allocation Mode	Use Case
physical-domain	1-1000	static	Bare metal, appliances, firewalls
VMM-domain	2001-3000	dynamic	Virtual environment
L3-domain	3091-3100	static	L3Outs

While this layout minimizes the number of VLAN pools and domains in ACI, it lacks granularity.

With the second type of VLAN pool and domain layout, central IT still manages everything in ACI, and management plane multitenancy is not seen as a business objective. However, there may be an orientation toward aligning VLAN pools and domains with function. [Table 6-3](#) shows an example of a layout that takes function into consideration.

Table 6-3 Single VLAN Pool per Function

Domain Name	VLAN Range	Allocation Mode	Use Case
physical-domain	1-900	static	Bare metal
firewall-domain	901-910	static	Trunks to firewalls
VMM-PROD	2001-2400	dynamic	vSphere production
VMM-NONPROD	2401-2800	dynamic	vSphere non-production
VMM-VOICE	2801-3000	dynamic	vSphere voice
L3core-domain	3091-3095	static	L3Outs to core layer
L3partner-domain	3096-3100	static	L3Outs to partner network

The layout illustrated in [Table 6-3](#) offers a lot more flexibility than the design outlined in [Table 6-2](#). For example, by separating VMM domains into the three separate vSphere environments, the organization has decided to align its domains with the function of each set of vCenter instances within the environment. This approach to domain definition provides administrators more flexibility in deploying EPGs solely to the desired vCenter environments.

Note

[Chapter 11](#), “[Integrating ACI into vSphere Using VDS](#),” covers vCenter and vSphere networking in detail. If the concepts presented on VMM domains in this chapter seem intimidating, come back and review this chapter once more after studying [Chapter 11](#).

As shown in [Table 6-3](#), a company may also want to allocate critical traffic such as firewalls and each L3Out into its own domain to reduce the impact of minor configuration mistakes.

[Table 6-4](#) shows an example of a granular layout that takes into consideration both function and tenancy. By creating dedicated domains for each tenant, the administrator defining the domains and pools is basically allocating certain VLAN ID ranges for dedicated use by specific tenants. Also, it is worth noting that VMM domains do not change in this example. This is because the dynamic VLAN allocation mode ensures that ACI itself (and not tenant administrators) is responsible for mapping VLANs to EPGs. This means separate per-tenant VLAN pools are not desired for VMM domains.

Table 6-4 A Hybrid Approach Oriented Toward Both Function and Tenancy

Domain Name	VLAN Range	Allocation Mode	Use Case
tenant-a-pdomain	1-280	static	Bare metal in Tenant A
tenant-a-l3domain	281-300	static	L3Outs in Tenant A
tenant-b-pdomain	301-580	static	Bare metal in Tenant B
tenant-b-l3domain	581-600	static	L3Outs in Tenant B
tenant-c-pdomain	601-880	static	Bare metal in Tenant C
tenant-c-l3domain	881-900	static	L3Outs in Tenant C

Domain Name	VLAN Range	Allocation Mode	Use Case
firewall-domain	901-910	static	Trunks to firewalls
VMM-PROD	2001-2400	dynamic	vSphere production
VMM-NONPROD	2401-2800	dynamic	vSphere non-production
VMM-VOICE	2801-3000	dynamic	vSphere voice

To create a domain, navigate to the **Fabric > Access Policies > Physical and External Domains** and select the folder related to the desired domain type. As shown in [Figure 6-4](#), you can right-click the Physical Domain folder and select Create Physical Domain to start the Create Physical Domain wizard.

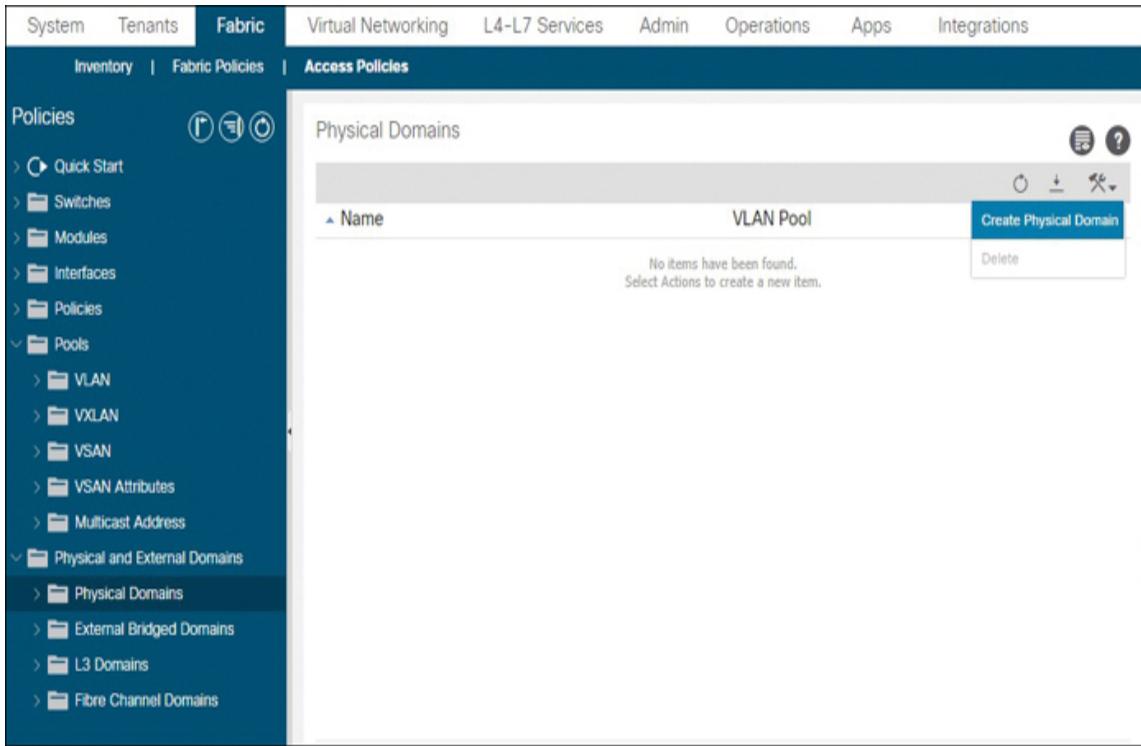


Figure 6-4 Opening the Create Physical Domain Wizard

In the Create Physical Domain wizard, type a name for the physical domain and select the VLAN pool that you want to associate with the new domain (see [Figure 6-5](#)). As a result of this configuration, any EPG that is able to bind to the domain called DCACI-Domain can potentially use VLAN IDs 910 through 920 as acceptable on-the-wire encapsulations.

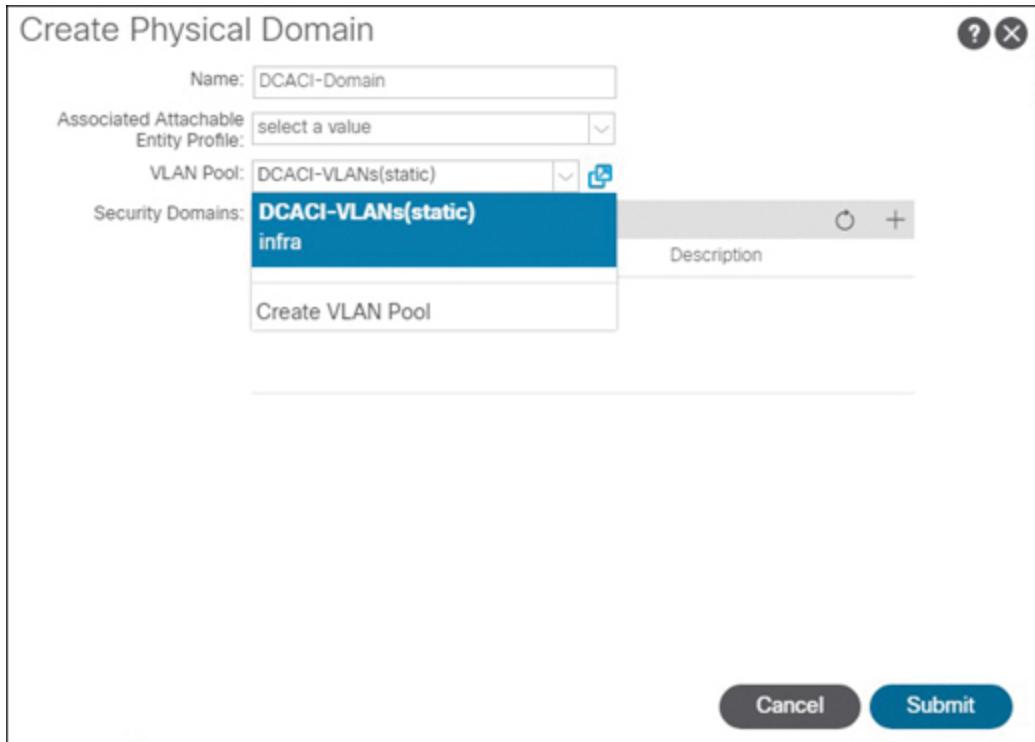


Figure 6-5 *Associating a VLAN Pool with a Domain*

Note

VMM domains cannot be created under Access Policies and need to be created under the Virtual Networking tab. The process for creating VMM domains is covered in [Chapter 11](#).

Challenges with Overlap Between VLAN Pools

Key Topic

Overlapping VLAN pools in ACI is not a problem in and of itself. It can become a problem, however, if an EPG has

associations with multiple domains and the domains reference overlapping VLAN pools.

Navigating the CLI to troubleshoot the resulting performance problems and traffic black-holing is beyond the scope of the DCACI 300-620 exam. However, it is still important to understand how you can sidestep this type of issue in the first place.

So how should VLAN pools and domains be created? In an ideal world, you should not need to overlap VLAN pools at all (as demonstrated in the examples of VLAN pools presented in the previous section).

In large environments, however, you may have or expect to have more than 4000 VLANs after all data center traffic is whitelisted. In such cases, you may have to plan for a lot more VLAN IDs than the number of VLANs currently in production. In such a case, you may want to dedicate switches for specific purposes so that overlapping VLAN pools and domains never fall onto the same set of leaf switches in the first place. [Table 6-5](#) presents an example of this type of VLAN pool and domain design.

Table 6-5 Optimizing VLAN Pools and Domains in Large Environments

Domain Name	VLAN Range	Allocation Mode	Leaf Node IDs
tenant-a-pdomain	1-900	static	101-110

tenant-a-l3domain	901-1000	static	101-110
tenant-b-pdomain	1001-1900	static	101-110
tenant-b-l3domain	1901-2000	static	101-110
tenant-c-pdomain	2001-2900	static	101-110
tenant-c-l3domain	2901-3000	static	101-110
firewall-domain	3001- 3100	static	101-110
VMM-VOICE	3101-4000	dynamic	101-110
VMM-PROD	1-4000	dynamic	111-120
VMM-NONPROD	1- 4000	dynamic	121-130

The design in [Table 6-5](#) makes several assumptions. First, it assumes that the environment in question has a very large virtual footprint consisting of a set of production vCenter instances and a set of non-production vCenter instances. Second, it assumes that the customer will not be pushing EPGs that are dedicated to production uses into vCenter instances that are meant for non-production use cases. Finally, it assumes that any given EPG will be assigned solely to a single domain.

Note

Despite the reasoning in the previous paragraph, multiple VMM domains can be linked to a single vCenter instance. The reason for this is that VMM domains are bound to data center objects in vCenter. Therefore, a vCenter instance that has multiple data center folders can have multiple VMM domain associations.

Can VXLAN be used instead of a VLAN to scale the number of segments in an environment beyond the 4094 usable VLAN limit that is common in traditional networks? The answer is yes! Certain virtual switches, such as AVS and AVE, leverage VXLAN. However, it is not very common for companies to want to install specialized drivers or specialized virtual switches in each server to enable support for VXLAN. That is why encapsulating traffic down to hypervisors and servers using VLAN IDs is still the norm with ACI. [Table 6-5](#) already showed how you can use good design practice to scale beyond the number of usable VLAN IDs within a single fabric. Another way ACI is able to use VXLAN internally to scale the number of segments in a fabric beyond what is possible with VLANs is through use of a feature called Port Local Scope, which is discussed in [Chapter 10](#), “Extending Layer 2 Outside ACI.”

Note

The concepts in this section are not documented best practices. The examples are meant solely to convey core concepts related to VLAN pools and domains. Furthermore, names of any objects should not be misconstrued as recommendations for naming best practices.

Attachable Access Entity Profiles (AAEPs)

So far, you have learned how to limit the VLAN IDs that can be used to encapsulate EPG traffic coming into and leaving an ACI fabric for each function and each endpoint attachment type. But how can a fabric administrator control where (for example, behind which switch ports, behind which vSphere servers) endpoints within each tenant can be deployed?



An **attachable access entity profile (AAEP)**, also referred to as an AEP, is a construct that fabric administrators use to authorize the placement of endpoint traffic on external entities, such as bare-metal servers, virtual machine hypervisors, switches, and routers. ACI can connect to external entities by using individual ports, port channels, or even vPCs.

To make this authorization possible, a user with the access-admin role or a role with equivalent privileges associates any number of domains, as needed, to an AAEP. Because any one port, port channel, or vPC configuration can

reference only a single AAEP, tenant administrators with access to a domain assigned to that AAEP are authorized to deploy endpoints behind the specified ports.

Just because a tenant administrator is authorized to deploy a server, a virtual machine, or another endpoint behind a switch port does not mean that the administrator is required to do so. Furthermore, the authorization provided by an AAEP does not actually provision VLANs on ports. Traffic for an EPG does not flow on ports until a tenant administrator maps the EPG to an encapsulation. The goal of an AAEP, therefore, is just to specify the potential scope of where endpoints associated with a domain are allowed to be deployed in the first place.

Note

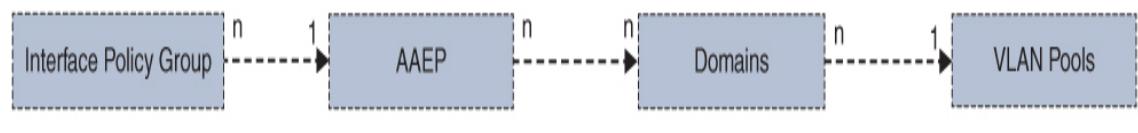
An AAEP EPG enables users with the access-admin role to map tenant EPGs to AAEPs directly from the Access Policies menu. AAEP EPGs are addressed in [Chapter 7, “Implementing Access Policies.”](#)

A tenant administrator who wants to map an EPG to a port needs to first bind the EPG to a domain. Through this domain association, the tenant administrator tells ACI how EPG endpoints are intended to connect to the fabric. Based on the VLAN pool the domain references, ACI knows which VLAN IDs are potential encapsulation options for the EPG. In any case, the tenant administrator is still limited to mapping the EPG to switch ports, port channels, or vPCs that reference an AAEP associated with the domain or domains bound to the EPG.

Note

A tenant administrator has visibility (access) to a domain only if the tenant administrator and domain have been assigned to the same security domain.

[Figure 6-6](#) illustrates the relationship between pools, domains, and AAEPs. In the first sample configuration in this figure, an engineer creates an AAEP called Infrastructure-AAEP with the domains Infra-Physical-Domain and Infra-VMM-Domain associated to it. The domain called Infra-Physical-Domain allows the attachment of bare-metal servers and appliances using any desired encapsulations between VLAN IDs 100 through 199. The VMM domain enables deployment of EPGs into a virtualized environment using any VLAN IDs between 2000 and 2999. The second example depicted provides a common configuration for L3Out domains where Layer 3 peerings may be established with adjacent devices over multiple VRFs or L3Outs via switch virtual interfaces (SVIs) over a single subset of physical ports. In this case, assume that Prod-L3Domain will be used for an L3Out in a VRF instance called Production, and NonProd-L3Domain will be used for an L3Out in a VRF called NonProduction. The Production L3Out SVIs in this case can use VLAN IDs 800 through 809, while the NonProduction L3Out SVIs can use VLAN IDs 810 through 819.



Abstract Reusable Port Configuration "Where" EPGs Can Connect "How" Endpoints Can Connect "Which" Encapsulation Can EPGs Use

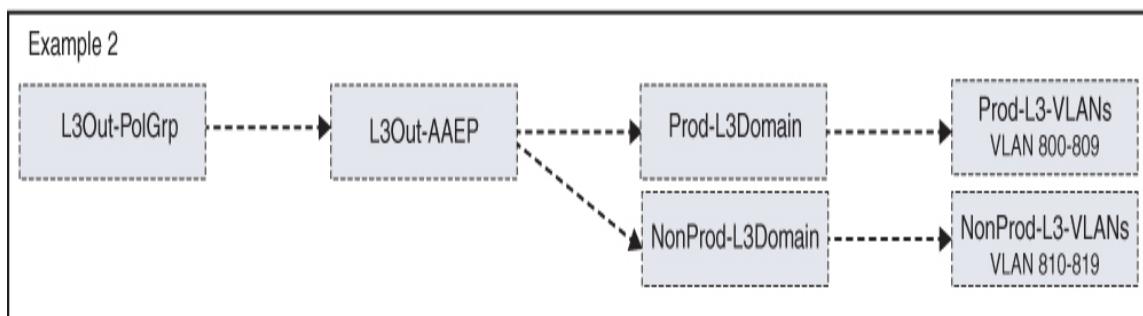
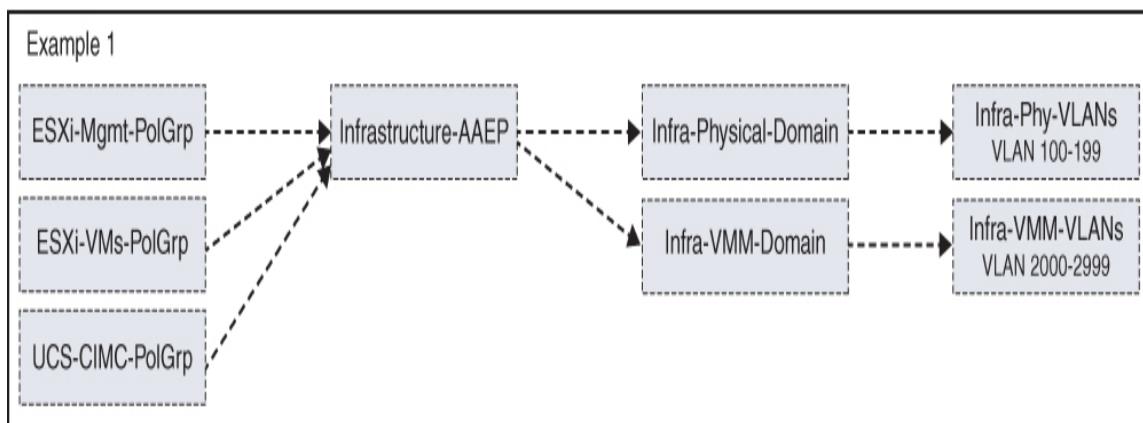


Figure 6-6 Pools, Domains, and AAEPs in Action

In [Figure 6-6](#), ESXi-Mgmt-PolGrp, ESXi-VMs-PolGrp, and UCS-CIMC-PolGrp point to AAEPs. (These objects are called *interface policy groups*, which have yet to be introduced.) To configure an AAEP, navigate to **Fabric > Access Policies > Policies > Global > Attachable Access Entity Profiles**. Right-click and select Create Attachable Access Entity Profile, as shown in [Figure 6-7](#).

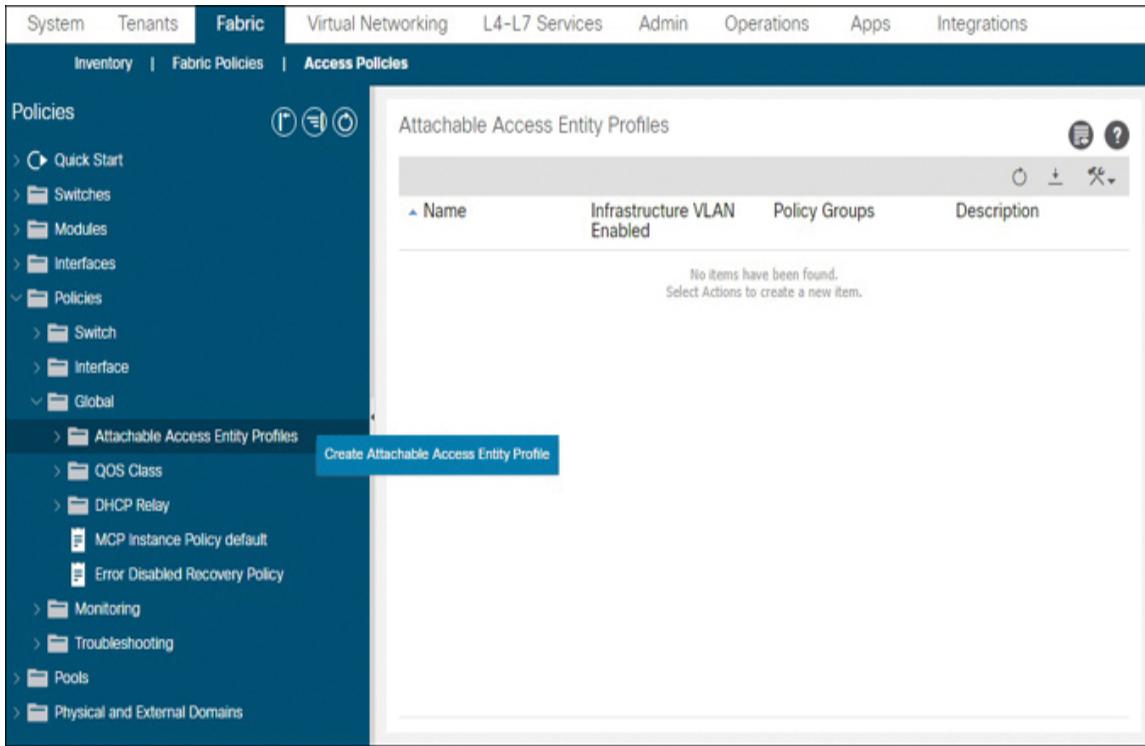


Figure 6-7 Navigating to the Create Attachable Access Entity Profile Wizard

In the Create Attachable Access Entity Profile wizard, type the desired AAEP name and select the domains you intend to associate with the AAEP. As domains are added to the AAEP, the acceptable encapsulation ranges dictated by the VLAN pools bound to each domain are displayed in the right column. When all the desired domains are selected and added to the AAEP, click Next. [Figure 6-8](#) shows the configuration of an AAEP named DCACI-AAEP with a physical domain association that allows the mapping of VLAN encapsulations 910 through 920.

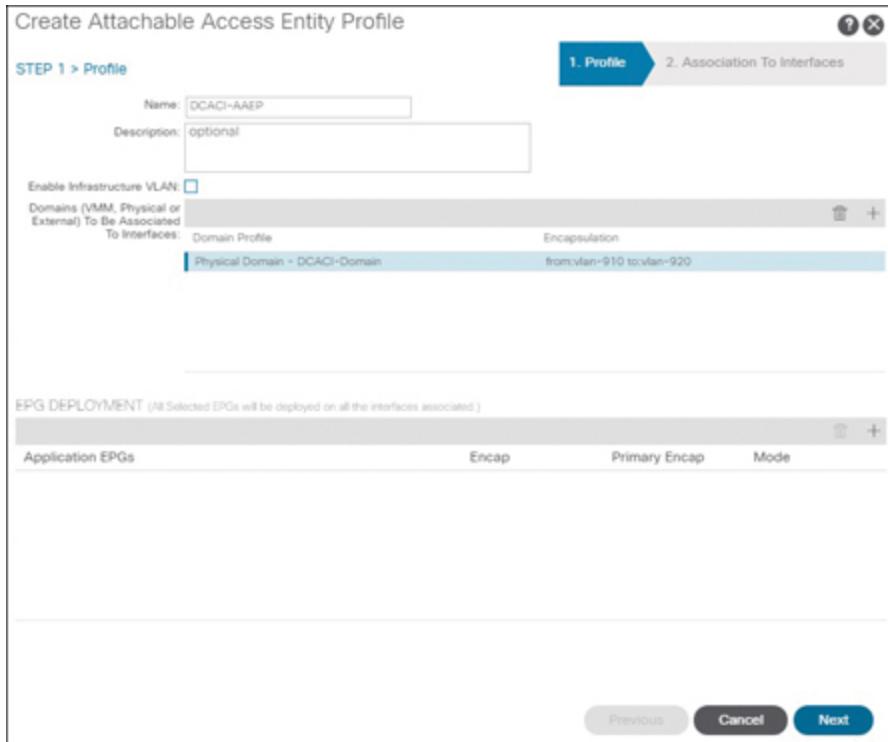


Figure 6-8 Creating an AAEP and Binding One or More Domains

If desired and if an interface policy group has been created, you can associate an AAEP with an interface policy group in the Create Attachable Access Entity Profile wizard. [Figure 6-9](#) shows the Association to Interfaces page of the wizard, but no interface policy groups have yet been configured. Note that it is more common to associate AAEPs to interface policy groups through the interface policy group configuration wizard. Click Finish to execute the AAEP creation.

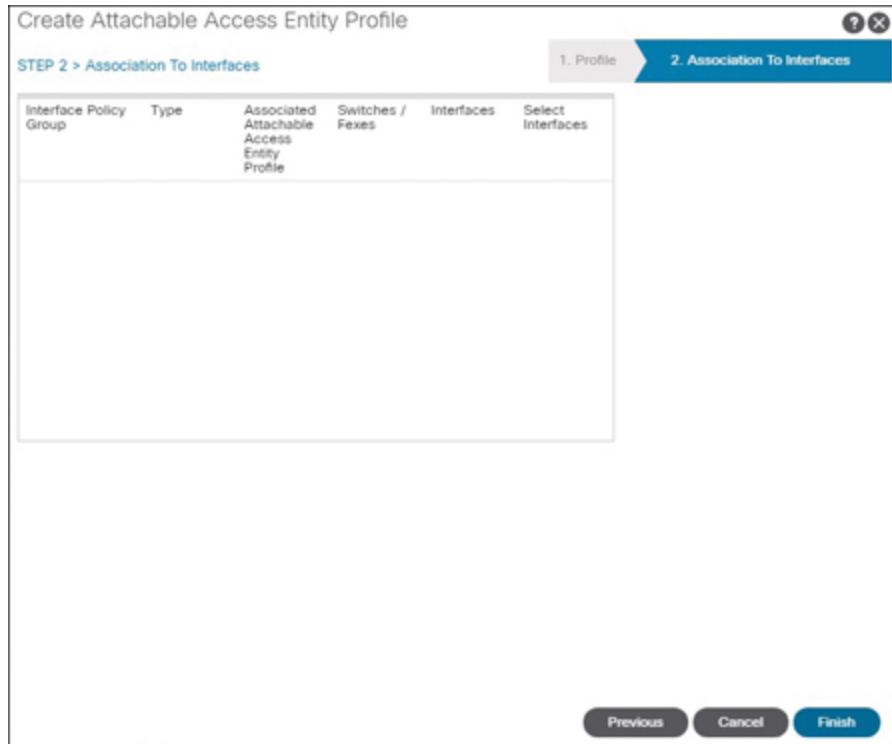


Figure 6-9 *Associating an AAEP to One or More Interface Policy Groups*

Note in [Figure 6-5](#), shown earlier in this chapter, that an AAEP was intentionally left unselected. From an object hierarchy perspective, a child object of a domain needs to reference an AAEP, and a child object of the AAEP needs to reference the domain to establish a bidirectional relationship. However, the configuration process shown in [Figures 6-7](#) through [6-9](#) creates all the required cross-references. [Figure 6-10](#) shows that DCACI-AAEP has been automatically associated with DCACI-Domain as a result of the AAEP to domain association shown earlier.