



Cisco APIC Troubleshooting Guide, Release 4.2(x)

First Published: 2019-09-08

Americas Headquarters

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



CONTENTS

PREFACE

Preface	xi
Audience	xi
Document Conventions	xii
Related Documentation	xiii
Documentation Feedback	xiii

CHAPTER 1

New and Changed	1
New and Changed Information	1

CHAPTER 2

Troubleshooting Overview	3
Troubleshooting Basics	4

CHAPTER 3

Troubleshooting APIC Crash Scenarios	7
Cisco APIC Cluster Failure Scenarios	7
Cluster Troubleshooting Scenarios	7
Cluster Faults	10
Troubleshooting Application Centric Infrastructure Crash Scenarios	12
Troubleshooting Fabric Node and Process Crash	12
APIC Process Crash Verification and Restart	13
Troubleshooting an APIC Process Crash	15

CHAPTER 4

Recovering Cisco APIC Passwords and Accessing Special Logins	17
Recovering the APIC Password	17
Using the Rescue-user Account to Erase the Cisco APIC Configuration Using the NX-OS Style CLI	18
Using the Fallback Login Domain to Log in to the Local Database	18

CHAPTER 5**Cisco APIC Troubleshooting Operations 21**

- Shutting Down the Cisco APIC System 21
- Shutting Down a Cisco APIC Using the GUI 21
- Using the APIC Reload Option Using the GUI 22
- Controlling the LED Locator Using the GUI 22

CHAPTER 6**Using the Cisco APIC Troubleshooting Tools 23**

- Enabling and Viewing ACL Contract and Deny Logs 24
 - About ACL Contract Permit and Deny Logs 24
 - Enabling ACL Contract Permit and Deny Logging Using the GUI 25
 - Enabling ACL Contract Permit Logging Using the NX-OS CLI 26
 - Enabling ACL Contract Permit Logging Using the REST API 26
 - Enabling Taboo Contract Deny Logging Using the GUI 27
 - Enabling Taboo Contract Deny Logging Using the NX-OS CLI 27
 - Enabling Taboo Contract Deny Logging Using the REST API 28
 - Viewing ACL Permit and Deny Logs Using the GUI 28
 - Viewing ACL Permit and Deny Logs Using the REST API 29
 - Viewing ACL Permit and Deny Logs Using the NX-OS CLI 30
- Using Atomic Counter Policies for Gathering Statistics 32
 - Atomic Counters 32
 - Atomic Counters Guidelines and Restrictions 33
 - Configuring Atomic Counters 34
 - Enabling Atomic Counters 35
 - Troubleshooting Using Atomic Counters with the REST API 36
- Enabling and Viewing Digital Optical Monitoring Statistics 36
 - Enabling Digital Optical Monitoring Using the GUI 36
 - Enabling Digital Optical Monitoring Using the REST API 37
 - Viewing Digital Optical Monitoring Statistics With the GUI 38
 - Troubleshooting Using Digital Optical Monitoring With the REST API 39
- Viewing and Understanding Health Scores 39
 - Health Score Types 40
 - Filtering by Health Score 40
 - Viewing Tenant Health 40

Viewing Fabric Health	40
Viewing MO Health in Visore	41
Debugging Health Scores Using Logs	41
Viewing Faults	41
Enabling Port Tracking for Uplink Failure Detection	42
Port Tracking Policy for Fabric Port Failure Detection	42
Configuring Port Tracking Using the GUI	43
Port Tracking Using the NX-OS CLI	43
Port Tracking Using the REST API	44
Configuring SNMP for Monitoring and Managing Devices	44
About SNMP	44
SNMP Access Support in Cisco ACI	45
Configuring the SNMP Policy Using the GUI	45
Configuring an SNMP Trap Destination Using the GUI	47
Configuring an SNMP Trap Source Using the GUI	48
Monitoring the System Using SNMP	48
Configuring SPAN for Traffic Monitoring	48
About SPAN	49
Multinode SPAN	49
SPAN Guidelines and Restrictions	50
Configuring SPAN Using the GUI	54
Configuring a Tenant SPAN Session Using the Cisco APIC GUI	54
Configuring a SPAN Filter Group Using the APIC GUI	54
Configuring an Access SPAN Policy Using the Cisco APIC GUI	55
Configuring a Fabric SPAN Policy Using the Cisco APIC GUI	56
Configuring a Layer 3 EPG SPAN Session for External Access Using the APIC GUI	57
Configuring a Destination Group for an Access SPAN Policy Using the Cisco APIC GUI	58
Configuring a Destination Group for a Fabric SPAN Policy Using the Cisco APIC GUI	58
Configuring SPAN Using the NX-OS Style CLI	59
Configuring Local SPAN in Access Mode	59
Configuring a SPAN Filter Group Using the NX-OS-Style CLI	61
Associating a SPAN Filter Group Using the NX-OS-Style CLI	63
Configuring ERSPAN in Access Mode	64
Configuring ERSPAN in Fabric Mode	68

Configuring ERSPAN in Tenant Mode	71
Configuring a Global SPAN-On-Drop Session Using the NX-OS-Style CLI	73
Configuring SPAN Using the REST API	74
Configuring a Fabric Destination Group for an ERSPAN Destination Using the REST API	74
Configuring a Global Drop Source Group Using the REST API	74
Configuring a Leaf Port as a SPAN Destination Using the REST API	75
Configuring a SPAN Access Source Group Using the REST API	75
Configuring a SPAN Fabric Source Group Using the REST API	75
Configuring an Access Destination Group for an ERSPAN Destination Using the REST API	76
Using Statistics	76
Viewing Statistics in the GUI	77
Switch Statistics Commands	77
Managing Statistics Thresholds Using the GUI	78
Statistics Troubleshooting Scenarios	79
Statistics Cleanup	80
Specifying Syslog Sources and Destinations	81
About Syslog	81
Creating a Syslog Destination and Destination Group	82
Creating a Syslog Source	83
Enabling Syslog to Display in NX-OS CLI Format, Using the REST API	84
Discovering Paths and Testing Connectivity with Traceroute	85
About Traceroute	85
About Windows and Linux Traceroute	85
Traceroute Guidelines and Restrictions	87
Performing a Traceroute Between Endpoints	88
Using the Troubleshooting Wizard	88
Getting Started with the Troubleshooting Wizard	89
Generating Troubleshooting Reports	91
Topology in the Troubleshooting Wizard	91
Using the Faults Troubleshooting Screen	93
Using the Drop/Statistics Troubleshooting Screen	94
Using the Contracts Troubleshooting Screen	96
Using the Events Troubleshooting Screen	97
Using the Traceroute Troubleshooting Screen	97

Using the Atomic Counter Troubleshooting Screen	99
Using the SPAN Troubleshooting Screen	99
Creating a SPAN Session Using the Cisco APIC Troubleshooting CLI	99
L4 - L7 Services Validated Scenarios	100
List of APIs for Endpoint to Endpoint Connections	101
interactive API	102
createsession API	103
modifysession API	104
atomiccounter API	104
traceroute API	104
span API	105
generatereport API	106
schedulingreport API	106
getreportstatus API	107
getreportslist API	107
getsessionslist API	107
getsessiondetail API	108
deletesession API	108
clearreports API	109
contracts API	109
List of APIs for Endpoint to Layer 3 External Connections	109
interactive API	110
createsession API	110
modifysession API	111
atomiccounter API	112
traceroute API	113
span API	114
generatereport API	114
schedulingreport API	116
getreportstatus API	117
getreportslist API	117
getsessionslist API	117
getsessiondetail API	118
deletesession API	119

- clearreports API 120
- contracts API 120
- ratelimit API 121
- l3ext API 121
- Checking for Configuration Synchronization Issues 122
- Viewing User Activities 122
 - Accessing User Activities 123
- Embedded Logic Analyzer Module 123
 - About the Embedded Logic Analyzer Module 123
 - Generating an ELAM Report in the Simplified Output for Modular Switches 123
 - Generating an ELAM Report in the Simplified Output for Fixed Form-Factor Switches 124

CHAPTER 7 Troubleshooting Switch Discovery 127

- Troubleshooting Switch Discovery Issues 127

CHAPTER 8 Manually Removing Disabled Interfaces and Decommissioned Switches from the GUI 131

- Manually Removing Disabled Interfaces and Decommissioned Switches from the GUI 131

CHAPTER 9 Decommissioning and Recommissioning Switches 133

- Decommissioning and Recommissioning Switches 133

CHAPTER 10 Troubleshooting Steps for Endpoint Connectivity Problems 135

- Troubleshooting Endpoint Connectivity 135
- Inspecting Endpoint and Tunnel Interface Status 136
 - Inspecting the Endpoint Status 136
 - Inspecting the Tunnel Interface Status 136
- Connecting an SFP Module 137

CHAPTER 11 Troubleshooting EVPN Type-2 Route Advertisement 139

- Troubleshooting EVPN Type-2 Route Distribution to a DCIG 139

CHAPTER 12 Performing a Rebuild of the Fabric 143

- Rebuilding the Fabric 143

CHAPTER 13	Verifying IP-Based EPG Configurations	145
	Verifying IP-Based EPG Configurations Using the GUI	145
	Verifying IP-EPG Configurations Using Switch Commands	146

CHAPTER 14	Recovering a Disconnected Leaf	149
	Recovering a Disconnected Leaf Using the NX-OS-Style CLI	149
	Recovering a Disconnected Leaf Using the REST API	150

CHAPTER 15	Troubleshooting a Loopback Failure	151
	Identifying a Failed Line Card	151

CHAPTER 16	Determining Why a PIM Interface Was Not Created	153
	A PIM Interface Was Not Created For an L3Out Interface	153
	A PIM Interface Was Not Created For a Multicast Tunnel Interface	154
	A PIM Interface Was Not Created For a Multicast-Enabled Bridge Domain	154

CHAPTER 17	Confirming the Port Security Installation	155
	Confirming Your Port Security Installation Using Visore	155
	Confirming Your Hardware Port Security Installation Using the Cisco NX-OS CLI	155

CHAPTER 18	Troubleshooting QoS Policies	159
	Troubleshooting Cisco APIC QoS Policies	159

CHAPTER 19	Determining the Supported SSL Ciphers	161
	About SSL Ciphers	161
	Determining the Supported SSL Ciphers Using the CLI	162

CHAPTER 20	Removing Unwanted _ui_ Objects	163
	Removing Unwanted _ui_ Objects Using the REST API	164

CHAPTER 21	Troubleshooting Multipod and Multi-Site Issues	165
	Troubleshooting Multi-Site and Multi-Pod	165

APPENDIX A **acidiag Command** **167**

APPENDIX B **Configuring Export Policies for Troubleshooting** **177**

- About Exporting Files **177**
- File Export Guidelines and Restrictions **177**
- Configuring a Remote Location **178**
 - Configuring a Remote Location Using the GUI **178**
 - Configuring a Remote Location Using the REST API **178**
 - Configuring a Remote Location Using the NX-OS Style CLI **178**
- Sending an On-Demand Tech Support File **180**
 - Sending an On-Demand Tech Support File Using the GUI **180**
 - Sending an On-Demand Tech Support File Using the REST API **180**

APPENDIX C **Finding the Switch Inventory** **183**

- Finding Your Switch Inventory Using the GUI **183**
- Finding Your Switch Inventory Using the NX-OS CLI **183**
- Finding Your Switch Inventory Using the REST API **186**

APPENDIX D **Cisco APIC SSD Replacement** **189**

- Replacing the Solid-State Drive in Cisco APIC **189**

APPENDIX E **Expected Output Errors** **191**

- Expected Output Errors **191**

APPENDIX F **Viewing CRC Error Counters** **193**

- Viewing CRC and Stomped CRC Error Counters **193**
- Viewing CRC Errors Using the GUI **193**
- Viewing CRC Errors Using the CLI **194**



Preface

This preface includes the following sections:

- [Audience, on page xi](#)
- [Document Conventions, on page xi](#)
- [Related Documentation, on page xiii](#)
- [Documentation Feedback, on page xiii](#)

Audience

This guide is intended for system and network engineers with a background in troubleshooting data systems, networks, and storage systems.

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
<i>Italic</i>	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.
[x {y z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Convention	Description
<i>variable</i>	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

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

This document uses the following conventions:



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



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



Warning IMPORTANT SAFETY INSTRUCTIONS

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

SAVE THESE INSTRUCTIONS

Related Documentation

Cisco Application Centric Infrastructure (ACI) Documentation

The ACI documentation is available at the following URL: <http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>.

Cisco Application Centric Infrastructure (ACI) Simulator Documentation

The Cisco ACI Simulator documentation is available at <http://www.cisco.com/c/en/us/support/cloud-systems-management/application-centric-infrastructure-simulator/tsd-products-support-series-home.html>.

Cisco Nexus 9000 Series Switches Documentation

The Cisco Nexus 9000 Series Switches documentation is available at <http://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/tsd-products-support-series-home.html>.

Cisco Application Virtual Switch Documentation

The Cisco Application Virtual Switch (AVS) documentation is available at <http://www.cisco.com/c/en/us/support/switches/application-virtual-switch/tsd-products-support-series-home.html>.

Cisco Application Centric Infrastructure (ACI) Integration with OpenStack Documentation

Cisco ACI integration with OpenStack documentation is available at <http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>.

Documentation Feedback

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



CHAPTER 1

New and Changed

- [New and Changed Information](#), on page 1

New and Changed Information

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

Cisco APIC Release Version	Feature	Description
4.2(1)	A new avread command was added. This command provides the same information as the acidiag avread command but in a tabular format.	acidiag Command , on page 167
	The <code>cluster_health</code> command was added to verify the cluster health.	<i>Verifying the Cisco APIC Cluster Using the CLI</i>
	The show discoveryissues command was added for troubleshooting switch discovery issues.	Troubleshooting Switch Discovery Issues , on page 127
	This release adds an option to the Embedded Logic Analyzer Module (ELAM) tool that changes the output to a human-readable format, which enables you to find key information quickly and more efficiently. In addition, hexadecimal values have been converted to decimal values in some instances for improved readability. For backward compatibility, the existing usage of ELAM is kept intact.	Generating an ELAM Report in the Simplified Output for Fixed Form-Factor Switches , on page 124 Generating an ELAM Report in the Simplified Output for Modular Switches , on page 123



CHAPTER 2

Troubleshooting Overview

The chapters in this guide describe common troubleshooting tips for specific Cisco APIC features and provide information about monitoring tools you can use for troubleshooting problems.

The features, issues, and tasks covered in this guide are listed below.

- **_ui_ Objects**—Explains how to remove unwanted **_ui_** objects caused by making changes with the **Basic Mode** or the NX-OS CLI before using the **Advanced Mode**.
- **acidiag**—Explains how to use the **acidiag** command for troubleshooting operations on the Cisco APIC.
- **Cisco APIC Cluster**—Explains how to diagnose cluster faults and troubleshoot common cluster issues. For basic cluster management information, see the appendix of this guide.
- **Cisco APIC Password Recovery and Emergency/Hidden Login Access**—Explains how to recover a password, how to access the rescue-user login to run troubleshooting commands, including erasing the configuration, and how to access a hidden login domain in case of a lockout.
- **Cisco APIC Troubleshooting Operations**—Explains how to gather information about your switches and how perform troubleshooting operations such as shutting down the system, shutting down the Cisco APIC controller, reloading the APIC controller, and turning on the LED locator.
- **Cisco APIC Troubleshooting Tools**—Explains how to use the Cisco APIC troubleshooting tools for debugging, monitoring traffic, viewing user activity history, checking for delays in synchronizing the policy manager and the policy distributor, and detecting issues such as traffic drops, misrouting, blocked paths, and uplink failures.
- **Endpoint Connectivity**—Explains how to troubleshoot endpoint connectivity using the Cisco APIC troubleshooting tools, such as traceroute, atomic counters, and SPAN, and how to connect an SFP module to a new card.



Note Information about the Cisco APIC troubleshooting tools is located in the [Using the Cisco APIC Troubleshooting Tools, on page 23](#) chapter.

- **EVPN Type-2 Host Routes**—Provides verification steps for this feature.
- **Export Policies**—Enables you to export statistics, tech support collections, faults and events, and to process core files and debug data from the fabric to any external hosts in a variety of formats.
- **Fabric Rebuild**—Explains how to rebuild your fabric.

- **Identifying a Failed Line Card**—Explains the procedure for identifying a line card that may have caused a loopback failure.
 - **IP-Based EPG**—Explains how to verify that you have correctly configured an IP-based EPG using the Cisco APIC GUI and using switch commands.
 - **Leaf Connectivity**—Explains how to recover a disconnected leaf using the REST API.
 - **PIM Interfaces**—Explains what to check when a PIM interface is not created for an L3Out, a multicast tunnel interface, or for a multicast-enabled bridge domain.
 - **Port Security**—Explains how to confirm your port security hardware and software installations.
 - **Removing Disabled Interfaces and Decommissioned Switches**—Explains how remove a disabled port entry in the GUI.
 - **Decommissioning and Recommissioning Switches**—Explains how to decommission and recommission nodes in a pod. A use case for this task would be to renumber the nodes in the pod in a more logical, scalable numbering convention.
 - **Cisco APIC SSD Replacement**—Explains how remove an SSD in the GUI.
 - **QoS**—Provides specific troubleshooting scenarios for this feature.
 - **SSL Ciphers**—Explains how to determine if an SSL cipher is supported.
 - **Switch Inventory**—Explains how to find the switch serial and model numbers. This helps TAC troubleshoot issues that you may experience.
 - **Expected Output Errors**—Provides an example of expected output errors observed from the internal counter interface of uplinks of the Cisco Nexus 93180YC-EX and ACI 93180YC-FX leaf switches in ACI mode.
- [Troubleshooting Basics, on page 4](#)

Troubleshooting Basics

The following are basic steps for troubleshooting:

Before you begin

- Familiarize yourself with the tools listed in [Using the Cisco APIC Troubleshooting Tools, on page 23](#).
- Familiarize yourself with the [Cisco APIC Troubleshooting Operations, on page 21](#).
- For issues with a specific feature, check the main contents of this guide for your feature. Troubleshooting tips are listed per-feature.

Step 1 Gather information that defines the specific symptoms.

Note In many cases, you can use the tools listed and described in the [Using the Cisco APIC Troubleshooting Tools, on page 23](#) chapter to gather useful troubleshooting information.

Step 2 Identify all potential problems that could be causing the symptoms.

Step 3 Systematically eliminate each potential problem (from most likely to least likely) until the symptoms disappear.

Note This guide provides step-by-step instructions for confirming the installation and configuration of specific features such as port security, endpoint connectivity, PIM, and IP-based EPGs. Following the instructions can help you to narrow down and resolve problems you are experiencing.



CHAPTER 3

Troubleshooting APIC Crash Scenarios

This chapter contains information about various failure or crash scenarios and possible recovery solutions.

This chapter contains the following sections:

- [Cisco APIC Cluster Failure Scenarios, on page 7](#)
- [Troubleshooting Application Centric Infrastructure Crash Scenarios, on page 12](#)

Cisco APIC Cluster Failure Scenarios

Cluster Troubleshooting Scenarios

The following table summarizes common cluster troubleshooting scenarios for the Cisco APIC.

Problem	Solution
An APIC node fails within the cluster. For example, node 2 of a cluster of 5 APICs fails.	<p>There are two available solutions:</p> <ul style="list-style-type: none">• Leave the target size and replace the APIC.• Reduce the cluster size to 4, decommission controller 5, and recommission it as APIC 2. The target size remains 4, and the operational size is 4 when the reconfigured APIC becomes active. <p>Note You can add a replacement APIC to the cluster and expand the target and operational size. For instructions on how to add a new APIC, refer to the <i>Cisco APIC Management, Installation, Upgrade, and Downgrade Guide</i>.</p>

Problem	Solution
<p>A new APIC connects to the fabric and loses connection to a leaf switch.</p>	<p>Use the following commands to check for an infra (infrastructure) VLAN mismatch:</p> <ul style="list-style-type: none"> • <code>cat /mit/sys/lldp/inst/if-[eth1--1]/ctrlradj/summary</code>—Displays the VLAN configured on the leaf switch. • <code>cat /mit/sys/lldp/inst/if-[eth1--1]/ctrlradj/summary</code>—Displays the infra (infrastructure) VLANs advertised by connected APICs. <p>If the output of these commands shows different VLANs, the new APIC is not configured with the correct infra (infrastructure) VLAN. To correct this issue, follow these steps:</p> <ul style="list-style-type: none"> • Log in to the APIC using <code>rescue-user</code>. <p>Note Admin credentials do not work because the APIC is not part of the fabric.</p> <ul style="list-style-type: none"> • Erase the configuration and reboot the APIC using the acdiag touch setup command. • Reconfigure the APIC. Verify that the fabric name, TEP addresses, and infra (infrastructure) VLAN match the APICs in the cluster. • Reload the leaf node.
<p>Two APICs cannot communicate after a reboot.</p>	<p>The issue can occur after the following sequence of events:</p> <ul style="list-style-type: none"> • APIC1 and APIC2 discover each other. • APIC1 reboots and becomes active with a new ChassisID (APIC1a) • The two APICs no longer communicate. <p>In this scenario, APIC1a discovers APIC2, but APIC2 is unavailable because it is in a cluster with APIC1, which appears to be offline. As a result, APIC1a does not accept messages from APIC2.</p> <p>To resolve the issue, decommission APIC1 on APIC2, and commission APIC1 again.</p>
<p>A decommissioned APIC joins a cluster.</p>	<p>The issue can occur after the following sequence of events:</p> <ul style="list-style-type: none"> • A member of the cluster becomes unavailable or the cluster splits. • An APIC is decommissioned. • After the cluster recovers, the decommissioned APIC is automatically commissioned. <p>To resolve the issue, decommission the APIC after the cluster recovers.</p>

Problem	Solution
Mismatched ChassisID following reboot.	<p>The issue occurs when an APIC boots with a ChassisID different from the ChassisID registered in the cluster. As a result, messages from this APIC are discarded.</p> <p>To resolve the issue, ensure that you decommission the APIC before rebooting.</p>
The APIC displays faults during changes to cluster size.	<p>A variety of conditions can prevent a cluster from extending the OperationalClusterSize to meet the AdministrativeClusterSize. For more information, inspect the fault and review the "Cluster Faults" section in the <i>Cisco APIC Basic Configuration Guide</i>.</p>
An APIC is unable to join a cluster.	<p>The issue occurs when two APICs are configured with the same ClusterID when a cluster expands. As a result, one of the two APICs cannot join the cluster and displays an expansion-contender-chassis-id-mismatch fault.</p> <p>To resolve the issue, configure the APIC outside the cluster with a new cluster ID.</p>
APIC unreachable in cluster.	<p>Check the following settings to diagnose the issue:</p> <ul style="list-style-type: none"> • Verify that fabric discovery is complete. • Identify the switch that is missing from the fabric. • Check whether the switch has requested and received an IP address from an APIC. • Verify that the switch has loaded a software image. • Verify how long the switch has been active. • Verify that all processes are running on the switch. For more information, see the "acidiag Command" section in the <i>Cisco APIC Basic Configuration Guide</i>. • Confirm that the missing switch has the correct date and time. • Confirm that the switch can communicate with other APICs.

Problem	Solution
Cluster does not expand.	<p>The issue occurs under the following circumstances:</p> <ul style="list-style-type: none"> • The OperationalClusterSize is smaller than the number of APICs. • No expansion contender (for example, the admin size is 5 and there is not an APIC with a clusterID of 4. • There is no connectivity between the cluster and a new APIC • Heartbeat messages are rejected by the new APIC • System is not healthy. • An unavailable appliance is carrying a data subset that is related to relocation. • Service is down on an appliance with a data subset that is related to relocation. • Unhealthy data subset related to relocation.
An APIC is down.	<p>Check the following:</p> <ul style="list-style-type: none"> • Connectivity issue—Verify connectivity using ping. • Interface type mismatch—Confirm that all APICs are set to in-band communication. • Fabric connectivity—Confirm that fabric connectivity is normal and that fabric discovery is complete. • Heartbeat rejected—Check the fltInfraIICIMsgSrcOutsider fault. Common errors include operational cluster size, mismatched ChassisID, source ID outside of the operational cluster size, source not commissioned, and fabric domain mismatch.

Cluster Faults

The APIC supports a variety of faults to help diagnose cluster problems. The following sections describe the two major cluster fault types.

Discard Faults

The APIC discards cluster messages that are not from a current cluster peer or cluster expansion candidate. If the APIC discards a message, it raises a fault that contains the originating APIC's serial number, cluster ID, and a timestamp. The following table summarizes the faults for discarded messages:

Fault	Meaning
expansion-contender-chassis-id-mismatch	The ChassisID of the transmitting APIC does not match the ChassisID learned by the cluster for expansion.
expansion-contender-fabric-domain-mismatch	The FabricID of the transmitting APIC does not match the FabricID learned by the cluster for expansion.

Fault	Meaning
expansion-contender-id-is-not-next-to-oper-cluster-size	The transmitting APIC has an inappropriate cluster ID for expansion. The value should be one greater than the current OperationalClusterSize.
expansion-contender-message-is-not-heartbeat	The transmitting APIC does not transmit continuous heartbeat messages.
fabric-domain-mismatch	The FabricID of the transmitting APIC does not match the FabricID of the cluster.
operational-cluster-size-distance-cannot-be-bridged	The transmitting APIC has an OperationalClusterSize that is different from that of the receiving APIC by more than 1. The receiving APIC rejects the request.
source-chassis-id-mismatch	The ChassisID of the transmitting APIC does not match the ChassisID registered with the cluster.
source-cluster-id-illegal	The transmitting APIC has a clusterID value that is not permitted.
source-has-mismatched-target-chassis-id	The target ChassisID of the transmitting APIC does not match the Chassis ID of the receiving APIC.
source-id-is-outside-operational-cluster-size	The transmitting APIC has a cluster ID that is outside of the OperationalClusterSize for the cluster.
source-is-not-commissioned	The transmitting APIC has a cluster ID that is currently decommissioned in the cluster.

Cluster Change Faults

The following faults apply when there is an error during a change to the APIC cluster size.

Fault	Meaning
cluster-is-stuck-at-size-2	This fault is issued if the OperationalClusterSize remains at 2 for an extended period. To resolve the issue, restore the cluster target size.
most-right-appliance-remains-commissioned	The last APIC within a cluster is still in service, which prevents the cluster from shrinking.
no-expansion-contender	The cluster cannot detect an APIC with a higher cluster ID, preventing the cluster from expanding.
service-down-on-appliance-carrying-replica-related-to-relocation	The data subset to be relocated has a copy on a service that is experiencing a failure. Indicates that there are multiple such failures on the APIC.
unavailable-appliance-carrying-replica-related-to-relocation	The data subset to be relocated has a copy on an unavailable APIC. To resolve the fault, restore the unavailable APIC.
unhealthy-replica-related-to-relocation	The data subset to be relocated has a copy on an APIC that is not healthy. To resolve the fault, determine the root cause of the failure.

APIC Unavailable

The following cluster faults can apply when an APIC is unavailable:

Fault	Meaning
fltInfraReplicaReplicaState	The cluster is unable to bring up a data subset.
fltInfraReplicaDatabaseState	Indicates a corruption in the data store service.
fltInfraServiceHealth	Indicates that a data subset is not fully functional.
fltInfraWiNodeHealth	Indicates that an APIC is not fully functional.

Troubleshooting Application Centric Infrastructure Crash Scenarios

Troubleshooting Fabric Node and Process Crash

The ACI switch node has numerous processes which control various functional aspects on the system. If the system has a software failure in a particular process, a core file will be generated and the process will be reloaded.

If the process is a Data Management Engine (DME) process, the DME process will restart automatically. If the process is a non-DME process, it will not restart automatically and the switch will reboot to recover.

This section presents an overview of the various processes, how to detect that a process has cored, and what actions should be taken when this occurs

DME Processes

The essential processes running on an APIC can be found through the CLI. Unlike the APIC, the processes that can be seen via the GUI in **FABRIC > INVENTORY > Pod 1 > node** shows all processes running on the leaf.

Through the **ps -ef | grep svc_ifc**:

```
rtp_leaf1# ps -ef |grep svc_ifc
root 3990 3087 1 Oct13 ? 00:43:36 /isan/bin/svc_ifc_policyelem --x
root 4039 3087 1 Oct13 ? 00:42:00 /isan/bin/svc_ifc_eventmgr --x
root 4261 3087 1 Oct13 ? 00:40:05 /isan/bin/svc_ifc_opflexelem --x -v
dptcp:8000
root 4271 3087 1 Oct13 ? 00:44:21 /isan/bin/svc_ifc_observerelem --x
root 4277 3087 1 Oct13 ? 00:40:42 /isan/bin/svc_ifc_dbgrelem --x
root 4279 3087 1 Oct13 ? 00:41:02 /isan/bin/svc_ifc_confelem --x
rtp_leaf1#
```

Each of the processes running on the switch writes activity to a log file on the system. These log files are bundled as part of the techsupport file but can be found via CLI access in /tmp/logs/ directory. For example, the Policy Element process log output is written into /tmp/logs/svc_ifc_policyelem.log.

The following is a brief description of the DME processes running on the system. This can help in understanding which log files to reference when troubleshooting a particular process or understand the impact to the system if a process crashed:

Process	Function
policyelem	Policy Element: Process logical MO from APIC and push concrete model to the switch
eventmgr	Event Manager: Processes local faults, events, health score
opflexelem	Opflex Element: Opflex server on switch
observerelem	Observer Element: Process local stats sent to APIC
dbgrelem	Debugger Element: Core handler
nginx	Web server handling traffic between the switch and APIC

Identify When a Process Crashes

When a process crashes and a core file is generated, a fault as well as an event is generated. The fault for the particular process is shown as a "process-crash" as shown in this syslog output from the APIC:

```
Oct 16 03:54:35 apic3 %LOG_LOCAL7-3-SYSTEM_MSG [E4208395][process-crash][major]
[subj-[dbg/cores/node-102-card-1-svc-policyelem-ts-2014-10-16T03:54:55.000+00:00]/
rec-12884905092]Process policyelem cored
```

When the process on the switch crashes, the core file is compressed and copied to the APIC. The syslog message notification comes from the APIC.

The fault that is generated when the process crashes is cleared when the process is Troubleshooting Cisco Application Centric Infrastructure 275 restarted. The fault can be viewed via the GUI in the fabric history tab at **FABRIC > INVENTORY > Pod 1**.

Collecting the Core Files

The APIC GUI provides a central location to collect the core files for the fabric nodes.

An export policy can be created from **ADMIN > IMPORT/EXPORT > Export Policies > Core**. However, there is a default core policy where files can be downloaded directly.

The core files can be accessed via SSH/SCP through the APIC at /data/techsupport on the APIC where the core file is located. Note that the core file will be available at /data/ techsupport on one APIC in the cluster, the exact APIC that the core file resides can be found by the Export Location path as shown in the GUI. For example, if the Export Location begins with "files/3/", the file is located on node 3 (APIC3).

APIC Process Crash Verification and Restart

Symptom 1

Process on switch fabric crashes. Either the process restarts automatically or the switch reloads to recover.

- **Verification:**

As indicated in the overview section, if a DME process crashes, it should restart automatically without the switch restarting. If a non-DME process crashes, the process will not automatically restart and the switch will reboot to recover.

Depending on which process crashes, the impact of the process core will vary.

When a non-DME process crashes, this will typically lead to a HAP reset as seen on the console:

```
[ 1130.593388] nvram_klm wrote rr=16 rr_str=ntp hap reset to nvram
[ 1130.599990] obfl_klm writing reset reason 16, ntp hap reset
[ 1130.612558] Collected 8 ext4 filesystems
```

- **Check Process Log:**

The process which crashes should have at some level of log output prior to the crash. The output of the logs on the switch are written into the /tmp/logs directory. The process name will be part of the file name. For example, for the Policy Element process, the file is svc_ifc_policyelem.log

```
rtp_leaf2# ls -l |grep policyelem
-rw-r--r-- 2 root root 13767569 Oct 16 00:37 svc_ifc_policyelem.log
-rw-r--r-- 1 root root 1413246 Oct 14 22:10 svc_ifc_policyelem.log.1.gz
-rw-r--r-- 1 root root 1276434 Oct 14 22:15 svc_ifc_policyelem.log.2.gz
-rw-r--r-- 1 root root 1588816 Oct 14 23:12 svc_ifc_policyelem.log.3.gz
-rw-r--r-- 1 root root 2124876 Oct 15 14:34 svc_ifc_policyelem.log.4.gz
-rw-r--r-- 1 root root 1354160 Oct 15 22:30 svc_ifc_policyelem.log.5.gz
-rw-r--r-- 2 root root 13767569 Oct 16 00:37 svc_ifc_policyelem.log.6
-rw-rw-rw- 1 root root 2 Oct 14 22:06 svc_ifc_policyelem.log.PRESERVED
-rw-rw-rw- 1 root root 209 Oct 14 22:06 svc_ifc_policyelem.log.stderr
rtp_leaf2#
```

There will be several files for each process located at /tmp/logs. As the log file increases in size, it will be compressed and older log files will be rotated off. Check the core file creation time (as shown in the GUI and the core file name) to understand where to look in the file. Also, when the process first attempts to come up, there be an entry in the log file that indicates “Process is restarting after a crash” that can be used to search backwards as to what might have happened prior to the crash.

- **Check Activity:**

A process which has been running has had some change which then caused it to crash. In many cases the changes may have been some configuration activity on the system. What activity occurred on the system can be found in the audit log history of the system.

- **Contact TAC:**

A process crashing should not normally occur. In order to understand better why beyond the above steps it will be necessary to decode the core file. At this point, the file will need to be collected and provided to the TAC for further processing.

Collect the core file (as indicated above how to do this) and open up a case with the TAC.

Symptom 2

Fabric switch continuously reloads or is stuck at the BIOS loader prompt.

- **Verification:**

If a DME process crashes, it should restart automatically without the switch restarting. If a non-DME process crashes, the process will not automatically restart and the switch will reboot to recover. However in either case if the process continuously crashes, the switch may get into a continuous reload loop or end up in the BIOS loader prompt.

```
[ 1130.593388] nvram_klm wrote rr=16 rr_str=policyelem hap reset to nvram
[ 1130.599990] obfl_klm writing reset reason 16, policyelem hap reset
[ 1130.612558] Collected 8 ext4 filesystems
```

• **Break the HAP Reset Loop:**

First step is to attempt to get the switch back into a state where further information can be collected.

If the switch is continuously rebooting, when the switch is booting up, break into the BIOS loader prompt through the console by typing CTRL C when the switch is first part of the boot cycle.

Once the switch is at the loader prompt, enter in the following commands:

- cmdline no_hap_reset
- boot

The cmdline command will prevent the switch from reloading with a hap reset is called. The second command will boot the system. Note that the boot command is needed instead of a reload at the loader as a reload will remove the cmdline option entered.

Though the system should now remain up to allow better access to collect data, whatever process is crashing will impact the functionality of the switch.

As in the previous table, check the process log, activity, and contact TAC steps.

Troubleshooting an APIC Process Crash

The APIC has a series of Data Management Engine (DME) processes which control various functional aspects on the system. When the system has a software failure in a particular process, a core file will be generated and the process will be reloaded.

The following sections cover potential issues involving system processes crashes or software failures, beginning with an overview of the various system processes, how to detect that a process has cored, and what actions should be taken when this occurs. The displays taken on a working healthy system can then be used to identify processes that may have terminated abruptly.

DME Processes

The essential processes running on an APIC can be found either through the GUI or the CLI. Using the GUI, the processes and the process ID running is found in **System > Controllers > Processes**.

Using the CLI, the processes and the process ID are found in the summary file at /aci/system/controllers/1/processes (for APIC1):

```
admin@RTP_Apic1:processes> cat summary
processes:
process-id process-name max-memory-allocated state
-----
0 KERNEL 0 interruptible-sleep
331 dhcpd 108920832 interruptible-sleep
336 vmmngr 334442496 interruptible-sleep
554 neo 398274560 interruptible-sleep
1034 ae 153690112 interruptible-sleep
1214 eventmgr 514793472 interruptible-sleep
2541 bootmgr 292020224 interruptible-sleep
4390 snoopy 28499968 interruptible-sleep
5832 scripthandler 254308352 interruptible-sleep
19204 dbgrr 648941568 interruptible-sleep
21863 nginx 4312199168 interruptible-sleep
32192 appliancedirector 136732672 interruptible-sleep
32197 sshd 1228800 interruptible-sleep
32202 perfwatch 19345408 interruptible-sleep
```

```
32203 observer 724484096 interruptible-sleep
32205 lldpad 1200128 interruptible-sleep
32209 topomgr 280576000 interruptible-sleep
32210 xinetd 99258368 interruptible-sleep
32213 policymgr 673251328 interruptible-sleep
32215 reader 258940928 interruptible-sleep
32216 logwatch 266596352 interruptible-sleep
32218 idmgr 246824960 interruptible-sleep
32416 keyhole 15233024 interruptible-sleep
admin@apic1:processes>
```

Each of the processes running on the APIC writes to a log file on the system. These log files can be bundled as part of the APIC techsupport file but can also be observed through SSH shell access in /var/log/dme/log. For example, the Policy Manager process log output is written into /var/log/dme/log/svc_ifc_policymgr.bin.log.

The following is a brief description of the processes running on the system. This can help in understanding which log files to reference when troubleshooting a particular process or understand the impact to the system if a process crashed:

Process	Function
KERNEL	Linux kernel
dhcpd	DHCP process running for APIC to assign infra addresses
vmmmgr	Handles process between APIC and Hypervisors
neo	Shell CLI Interpreter
ae	Handles the state and inventory of local APIC appliance
eventmgr	Handles all events and faults on the system
bootmgr	Controls boot and firmware updates on fabric nodes
snoopy	Shell CLI help, tab command completion
scripthandler	Handles the L4-L7 device scripts and communication
dbgr	Generates core files when process crashes
nginx	Web service handling GUI and REST API access
apliancedirector	Handles formation and control of APIC cluster
sshd	Enabled SSH access into the APIC
perfwatch	Monitors Linux cgroup resource usage
observer	Monitors the fabric system and data handling of state, stats, health
lldpad	LLDP Agent
topomgr	Maintains fabric topology and inventory



CHAPTER 4

Recovering Cisco APIC Passwords and Accessing Special Logins

This chapter explains how to recover your Cisco APIC password, how to access the rescue-user login to run troubleshooting commands, including the command for erasing the configuration, and how to access a hidden login domain that allows you to log in using the local user database in case of a lockout.

This chapter contains the following sections:

- [Recovering the APIC Password, on page 17](#)
- [Using the Rescue-user Account to Erase the Cisco APIC Configuration Using the NX-OS Style CLI, on page 18](#)
- [Using the Fallback Login Domain to Log in to the Local Database, on page 18](#)

Recovering the APIC Password

Follow these steps to recover the APIC password.

- Step 1** Create and save an empty file named "aci-admin-passwd-reset.txt".
 - Step 2** Add the file to a USB drive. You can format the USB drive to FAT or FAT32.
 - Step 3** Connect the USB drive to one of the rear USB ports on the Cisco APIC.
 - Step 4** Reboot the Cisco APIC using Cisco Integrated Management Controller (CIMC) or by hard power cycling the device.
 - Step 5** Press the **Esc** key during the 10-second countdown timer that appears at the top left to bring up the list of boot targets.
 - Step 6** Press the **e** key to edit the default grub line.
 - Step 7** Go to the line that begins with "linux." Using the **End** key or **Right Arrow** key, move the cursor to the end of that line and append "aci-admin-passwd-reset".
 - Step 8** Press **Ctrl+X** to boot the entry.
- It may take a few minutes for the new password to take effect.
-

Using the Rescue-user Account to Erase the Cisco APIC Configuration Using the NX-OS Style CLI

The rescue-user is an emergency login that provides access to the Cisco APIC even when it is not in a cluster. You can use this login to run troubleshooting commands including erasing the configuration.



Note For a standby Cisco APIC, you can log in using SSH with the username "rescue-user" and no password. If the standby Cisco APIC was previously part of a fabric, the "rescue-user" account will retain the old administrator password, unless the operating system is re-installed using the keyboard, video, mouse (KVM) console.

Step 1 Access the APIC using the Cisco Integrated Management Controller (CIMC) console.

Step 2 Login as rescue-user.

Note If an admin password is in place and the Cisco APIC is logged onto the fabric, the rescue-user password is the same as the admin password. Otherwise there is no rescue-user password.

Step 3 Use the **acidiag touch** command to clear the configuration.

Example:

```
apic1# acidiag touch setup
```

Using the Fallback Login Domain to Log in to the Local Database

There is a hidden login domain named "fallback" that allows you to log in using the local user database in case of lockout. The format of the username used for the authentication method is `apic#fallback\<username>`.

Step 1 Use the fallback login domain to log in to the local database in the GUI or log in to the fallback login domain using the NX-OS-style CLI, shown as follows:

```
apic1(config)# aaa authentication login domain fallback
apic1(config-domain)# ?
group Set provider group for login domain
realm Specify server realm
```

Step 2 Optionally, you can instead use the REST API to log in to the fallback login domain, shown as follows:

- URL: `https://ip_address/api/aaaLogin.xml`
- DATA:


```
<aaaUser name="apic#fallback\admin"  
pwd="passwordhere"/>
```



CHAPTER 5

Cisco APIC Troubleshooting Operations

This chapter explains how to perform the basic troubleshooting operations and contains the following sections:

- [Shutting Down the Cisco APIC System, on page 21](#)
- [Shutting Down a Cisco APIC Using the GUI, on page 21](#)
- [Using the APIC Reload Option Using the GUI, on page 22](#)
- [Controlling the LED Locator Using the GUI, on page 22](#)

Shutting Down the Cisco APIC System

This procedure shuts down the Cisco Application Policy Infrastructure Controller (APIC) system. After you shut down the system, you will relocate the entire fabric and power it up, then update the time zone and/or NTP servers accordingly.

Before you begin

Ensure cluster health is fully fit.

-
- Step 1** On the menu bar, choose **System > Controllers**.
 - Step 2** In the Navigation pane, choose **Controllers > apic_name**.
 - Step 3** Right-click the Cisco APIC and choose **Shutdown**.
 - Step 4** Relocate the Cisco APIC, then power it up.
 - Step 5** Confirm that the cluster has fully converged.
 - Step 6** Repeat this procedure for the next Cisco APIC.
-

Shutting Down a Cisco APIC Using the GUI

This procedure shuts down a Cisco Application Policy Infrastructure Controller (APIC). This procedure shuts down only one Cisco APIC, not the entire Cisco APIC system itself. Following this procedure causes the controller to shut down immediately. Use caution in performing a shutdown because the only way to bring the controller back up is to do so from the actual machine. If you need to access the machine, see [Controlling the LED Locator Using the GUI, on page 22](#).



Note If possible, move Cisco APICs one at a time. As long as there are at least two Cisco APICs in the cluster online, there is read/write access. If you need to relocate more than one Cisco APIC at a time, this results in one or no remaining controllers online, and the fabric will go into a read-only mode when they are shut down. During this time, there can be no policy changes including endpoint moves (which includes virtual machine movement).

-
- Step 1** On the menu bar, choose **System** > **Controllers**.
 - Step 2** In the Navigation pane, choose **Controllers** > *apic_name*.
 - Step 3** Right-click the Cisco APIC and choose **Shutdown**.
 - Step 4** Relocate the Cisco APIC, then power it up.
 - Step 5** Confirm that the cluster has fully converged.
-

Using the APIC Reload Option Using the GUI

This procedure reloads the Cisco Application Policy Infrastructure Controller (APIC), not the entire Cisco APIC system, using the GUI.

-
- Step 1** On the menu bar, choose **System** > **Controllers**.
 - Step 2** In the Navigation pane, choose **Controllers** > *apic_name*.
 - Step 3** Right-click the Cisco APIC and choose **Reload**.
-

Controlling the LED Locator Using the GUI

This procedure turns on or off the LED locator for the Cisco Application Policy Infrastructure Controller (APIC) using the GUI.

-
- Step 1** On the menu bar, choose **System** > **Controllers**.
 - Step 2** In the Navigation pane, choose **Controllers** > *apic_name*.
 - Step 3** Right-click the Cisco APIC and choose **Turn On Locator LED** or **Turn On Locator LED** as appropriate.
-



CHAPTER 6

Using the Cisco APIC Troubleshooting Tools

This chapter introduces the tools and methodology commonly used to troubleshoot problems you may experience. These tools can assist you with monitoring traffic, debugging, and detecting issues such as traffic drops, misrouting, blocked paths, and uplink failures. See the tools listed below for a summary overview of the tools described in this chapter:

- **ACL Contract Permit and Deny Logs**—Enables the logging of packets or flows that were allowed to be sent because of contract permit rules and the logging of packets or flows dropped because of taboo contract deny rules.
- **Atomic Counters**—Enables you to gather statistics about traffic between flows for detecting drops and misrouting in the fabric and for enabling quick debugging and isolation of application connectivity issues.
- **Digital Optical Monitoring**—Enables you to view digital optical monitoring (DOM) statistics about a physical interface.
- **Health Scores**—Enables you to isolate performance issues by drilling down through the network hierarchy to isolate faults to specific managed objects (MOs).
- **Port Tracking**—Enables you to monitor the status of links between leaf switches and spine switches for detecting uplink failure.
- **SNMP**—Simple Network Management Protocol (SNMP) enables you to remotely monitor individual hosts (APIC or another host) and find out the state of any particular node.
- **SPAN**—Switchport Analyzer (SPAN) enables you to perform detailed troubleshooting or to take a sample of traffic from a particular application host for proactive monitoring and analysis.
- **Statistics**—Provides real-time measures of observed objects. Viewing statistics enable you to perform trend analysis and troubleshooting.
- **Syslog**—Enables you to specify the minimum severity level of messages to be sent, the items to be included in the syslog messages, and the syslog destination. The format can also be displayed in NX-OS CLI format.
- **Traceroute**—Enables you to find the routes that packets actually take when traveling to their destination.
- **Troubleshooting Wizard**—Enables administrators to troubleshoot issues that occur during specific time frames, which can be designated by selecting two endpoints.
- **Configuration Sync Issues**—Enables you to see if any transactions in Cisco APIC have not yet synced.

This chapter contains the following sections:

- [Enabling and Viewing ACL Contract and Deny Logs, on page 24](#)
- [Using Atomic Counter Policies for Gathering Statistics, on page 32](#)
- [Enabling and Viewing Digital Optical Monitoring Statistics, on page 36](#)
- [Viewing and Understanding Health Scores, on page 39](#)

- [Enabling Port Tracking for Uplink Failure Detection, on page 42](#)
- [Configuring SNMP for Monitoring and Managing Devices, on page 44](#)
- [Configuring SPAN for Traffic Monitoring, on page 48](#)
- [Using Statistics, on page 76](#)
- [Specifying Syslog Sources and Destinations, on page 81](#)
- [Discovering Paths and Testing Connectivity with Traceroute, on page 85](#)
- [Using the Troubleshooting Wizard, on page 88](#)
- [Checking for Configuration Synchronization Issues, on page 122](#)
- [Viewing User Activities, on page 122](#)
- [Embedded Logic Analyzer Module, on page 123](#)

Enabling and Viewing ACL Contract and Deny Logs

About ACL Contract Permit and Deny Logs

To log and/or monitor the traffic flow for a contract rule, you can enable and view the logging of packets or flows that were allowed to be sent because of contract permit rules or the logging of packets or flows that were dropped because of:

- Taboo contract deny rules
- Deny actions in contract subjects
- Contract or subject exceptions
- ACL contract permit in the ACI fabric is only supported on Nexus 9000 Series switches with names that end in EX or FX, and all later models. For example, N9K-C93180LC-EX or N9K-C9336C-FX.
- Deny logging in the ACI fabric is supported on all platforms.
- Using log directive on filters in management contracts is not supported. Setting the log directive will cause zoning-rule deployment failure.

For information on standard and taboo contracts and subjects, see *Cisco Application Centric Infrastructure Fundamentals* and *Cisco APIC Basic Configuration Guide*.

EPG Data Included in ACL Permit and Deny Log Output

Up to Cisco APIC, Release 3.2(1), the ACL permit and deny logs did not identify the EPGs associated with the contracts being logged. In release 3.2(1) the source EPG and destination EPG are added to the output of ACI permit and deny logs. ACL permit and deny logs include the relevant EPGs with the following limitations:

- Depending on the position of the EPG in the network, EPG data may not be available for the logs.
- When configuration changes occur, log data may be out of date. In steady state, log data is accurate.

The most accurate EPG data in the permit and deny logs results when the logs are focussed on:

- Flows from EPG to EPG, where the ingress policy is installed at the ingress TOR and the egress policy is installed at the egress TOR.

- Flows from EPG to L3Out, where one policy is applied on the border leaf TOR and the other policy is applied on a non-BL TOR.

EPGs in the log output are not supported for uSeg EPGs or for EPGs used in shared services (including shared L3Outs).

Enabling ACL Contract Permit and Deny Logging Using the GUI

The following steps show how to enable contract permit and deny logging using the GUI:



Note The tenant that contains the permit logging is the tenant that contains the VRF that the EPG is associated to. This will not necessarily be the same tenant as the EPG or its associated contracts.

-
- Step 1** On the menu bar, choose **Tenants** > <tenant name>.
- Step 2** In the **Navigation** pane, expand **Contracts**, right-click **Standard**, and choose **Create Contract**.
- Step 3** In the **Create Contract** dialog box, perform the following actions:
- a) In the **Name** field, type the name for the contract.
 - b) In the **Scope** field, choose the scope for it (VRF, Tenant, or Global).
 - c) Optional. Set the target DSCP or QoS class to be applied to the contract.
 - d) Click the + icon to expand **Subjects**.
- Step 4** In the Create Contract Subject dialog box, perform the following actions:
- Step 5** Enter the name of the subject and an optional description.
- Step 6** Optional. From the drop-down list for the target DSCP, select the DSCP to be applied to the subject.
- Step 7** Leave **Apply Both Directions** checked, unless you want the contract to only be applied from the consumer to the provider, instead of in both directions.
- Step 8** Leave **Reverse Filter Ports** checked if you unchecked **Apply Both Directions** to swap the Layer 4 source and destination ports so that the rule is applied from the provider to the consumer.
- Step 9** Click the + icon to expand **Filters**.
- Step 10** In the **Name** drop-down list, choose an option; for example, click **arp**, **default**, **est**, or **icmp**, or choose a previously configured filter.
- Step 11** In the **Directives** drop-down list, click **log**.
- Step 12** (Optional) Change the Action to be taken with this subject to **Deny** (or leave the action to the default, **Permit**).
- With Directive: log enabled, if the action for this subject is **Permit**, ACL permit logs track the flows and packets that are controlled by the subject and contract. If the action for this subject is **Deny**, ACL deny logs track the flows and packets.
- Step 13** (Optional) Set the priority for the subject.
- Step 14** Click **Update**.
- Step 15** Click **OK**.
- Step 16** Click **Submit**.
Logging is enabled for this contract.
-

Enabling ACL Contract Permit Logging Using the NX-OS CLI

The following example shows how to enable Contract permit logging using the NX-OS CLI.

- Step 1** To enable logging of packets or flows that were allowed to be sent because of Contract permit rules, use the following commands:

```
configure
tenant <tenantName>
contract <contractName> type <permit>
subject <subject Name>
access-group <access-list> <in/out/both> log
```

Example:

For example:

```
apic1# configure
apic1(config)# tenant BDMoDel
apic1(config-tenant)# contract Logicmp type permit
apic1(config-tenant-contract)# subject icmp
apic1(config-tenant-contract-subj)# access-group arp both log
```

- Step 2** To disable the permit logging use the **no** form of the access-group command; for example, use the `no access-group arp both log` command.

Enabling ACL Contract Permit Logging Using the REST API

The following example shows you how to enable permit and deny logging using the REST API. This example configures ACL permit and deny logging for a contract with subjects that have Permit and Deny actions configured.

For this configuration, send a post with XML similar to the following example:

Example:

```
<vzBrCP dn="uni/tn-Tenant64/brc-C64" name="C64" scope="context">
  <vzSubj consMatchT="AtleastOne" name="HTTPSSbj" provMatchT="AtleastOne" revFltPorts="yes"
rn="subj-HTTPSSbj">
    <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes" priorityOverride="default"
rn="rssubjFiltAtt-PerHTTPS" tDn="uni/tn-Tenant64/flt-PerHTTPS" tRn="flt-PerHTTPS"
tnVzFilterName="PerHTTPS"/>
  </vzSubj>
  <vzSubj consMatchT="AtleastOne" name="httpSbj" provMatchT="AtleastOne" revFltPorts="yes"
rn="subj-httpSbj">
    <vzRsSubjFiltAtt action="deny" directives="log" forceResolve="yes" priorityOverride="default"
rn="rssubjFiltAtt-httpFilter" tDn="uni/tn-Tenant64/flt-httpFilter" tRn="flt-httpFilter"
tnVzFilterName="httpFilter"/>
  </vzSubj>
  <vzSubj consMatchT="AtleastOne" name="subj64" provMatchT="AtleastOne" revFltPorts="yes"
rn="subj-subj64">
    <vzRsSubjFiltAtt action="permit" directives="log" forceResolve="yes" priorityOverride="default"
rn="rssubjFiltAtt-icmp" tDn="uni/tn-common/flt-icmp" tRn="flt-icmp" tnVzFilterName="icmp"/>
```



```
</vzSubj>
</vzBrCP>
```

Enabling Taboo Contract Deny Logging Using the GUI

The following steps show how to enable Taboo Contract deny logging using the GUI.

-
- Step 1** On the menu bar, choose **Tenants** > <tenant name>.
- Step 2** In the **Navigation** pane, expand **Contracts**.
- Step 3** Right-click **Taboos** and choose **Create Taboo Contract**.
- Step 4** In the Create Taboo Contract dialog box, perform the following actions to specify the Taboo contract:
- In the **Name** field, type the name for the contract.
 - Optional. In the **Description** field, type a description of the Taboo contract.
 - Click the + icon to expand **Subjects**.
- Step 5** In the **Create Taboo Contract Subject** dialog box, perform the following actions:
- In the Specify Identity of Subject area, type a name and optional description.
 - Click the + icon to expand **Filters**.
 - From the **Name** drop-down list, choose one of the default values, such as <tenant_name>/arp, <tenant_name>/default, <tenant_name>/est, <tenant_name>/icmp, choose a previously created filter, or **Create Filter**.
- Note** If you chose **Create Filter**, in the Specify Filter Identity Area, perform the following actions to specify criteria for the ACL Deny rule:
- Type a name and optional description.
 - Expand **Entries**, type a name for the rule, and choose the criteria to define the traffic you want to deny.
 - In the **Directives** drop-down list, choose **log**.
 - Click **Update**.
 - Click **OK**.
- Step 6** Click **Submit**.
Logging is enabled for this Taboo contract.
-

Enabling Taboo Contract Deny Logging Using the NX-OS CLI

The following example shows how to enable Taboo Contract deny logging using the NX-OS CLI.

-
- Step 1** To enable logging of packets or flows dropped because of Taboo Contract deny rules, use the following commands:
- ```
configure
tenant <tenantName>
contract <contractName> type <deny>
```

```
subject <subject Name>
access-group <access-list> <both> log
```

**Example:**

For example:

```
apic1# configure
apic1(config)# tenant BDMoDel
apic1(config-tenant)# contract dropFTP type deny
apic1(config-tenant-contract)# subject dropftp
apic1(config-tenant-contract-subj)# access-group ftp both log
```

- Step 2** To disable the deny logging use the **no** form of the access-group command; for example, use the `no access-group https both log` command.

## Enabling Taboo Contract Deny Logging Using the REST API

The following example shows you how to enable Taboo Contract deny logging using the REST API.

To configure taboo contract deny logging, send a post with XML similar to the following example.

**Example:**

```
<vzTaboo dn="uni/tn-Tenant64/taboo-TCtrctPrefix" name="TCtrctPrefix" scope="context">
 <vzTSubj name="PrefSubj" rn="tsubj-PrefSubj">
 <vzRsDenyRule directives="log" forceResolve="yes" rn="rsdenyRule-default" tCl="vzFilter"
tDn="uni/tn-common/flt-default" tRn="flt-default"/>
 </vzTSubj>
</vzTaboo>
```

## Viewing ACL Permit and Deny Logs Using the GUI

The following steps show how to view ACL permit and deny logs (if they are enabled) for traffic flows, using the GUI:

- Step 1** On the menu bar, choose **Tenants** > <tenant name>.
- Step 2** In the **Navigation** pane, click on **Tenant** <tenant name>.
- Step 3** In the **Tenants** <tenant name> **Work** pane, click the **Operational** tab.
- Step 4** Under the **Operational** tab, click the **Flows** tab.  
Under the **Flows** tab, click one of the tabs to view log data for Layer 2 permit logs (**L2 Permit**) Layer 3 permit logs (**L3 Permit**), Layer 2 deny logs (**L2 Drop**), or Layer 3 deny logs (**L3 Drop**). On each tab, you can view ACL logging data, if traffic is flowing. The data points differ according to the log type and ACL rule; for example, the following data points are included for **L3 Permit** and **L3 Deny** logs:
- VRF
  - Alias
  - Source IP address

- Destination IP address
- Protocol
- Source port
- Destination port
- Source MAC address
- Destination MAC address
- Node
- Source interface
- VRF Encap
- Source EPG
- Destination EPG
- Source PC Tag
- Destination PC Tag

**Note** You can also use the **Packets** tab (next to the **Flows** tab) to access ACL logs for groups of packets (up to 10) with the same signature, source and destination. You can see what type of packets are being sent and which are being dropped.

---

## Viewing ACL Permit and Deny Logs Using the REST API

The following example shows how to view Layer 2 deny log data for traffic flows, using the REST API. You can send queries using the following MOs:

- `aclogDropL2Flow`
- `aclogPermitL2Flow`
- `aclogDropL3Flow`
- `aclogPermitL3Flow`
- `aclogDropL2Pkt`
- `aclogPermitL2Pkt`
- `aclogDropL3Pkt`
- `aclogPermitL3Pkt`

### Before you begin

You must enable permit or deny logging, before you can view ACL contract permit and deny log data.

To view Layer 3 drop log data, send the following query using the REST API:

```
GET https://apic-ip-address/api/class/acllogDropL3Flow
```

**Example:**

The following example shows sample output:

```
<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="2">
 <acllogPermitL3Flow childAction="" dn="topology/pod-1/node-101/ndbgs/acllog/tn-common/ctx-inb
/permitl3flow-spctag-333-dpctag-444-sepname-unknown-depname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]
-dip-[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-
[port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
dstMacAddr="00:00:12:00:00:25"
dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn="" protocol="udp"
srcEpgName="unknown"
srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0" srcMacAddr="00:00:15:00:00:28"
srcPcTag="333"
srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
 <acllogPermitL3Flow childAction="" dn="topology/pod-1/node-102/ndbgs/acllog/tn-common/ctx-inb
/permitl3flow-spctag-333-dpctag-444-sepname-unknown-depname-unknown-sip-[100:c000:a00:700:b00:0:f00:0]-dip-
[19.0.2.10]-proto-udp-sport-17459-dport-8721-smac-00:00:15:00:00:28-dmac-00:00:12:00:00:25-sintf-
[port-channel5]-vrfencap-VXLAN: 2097153" dstEpgName="unknown" dstIp="19.0.2.10"
dstMacAddr="00:00:12:00:00:25"
dstPcTag="444" dstPort="8721" lcOwn="local" modTs="never" monPolDn="" protocol="udp"
srcEpgName="unknown"
srcIntf="port-channel5" srcIp="100:c000:a00:700:b00:0:f00:0" srcMacAddr="00:00:15:00:00:28"
srcPcTag="333"
srcPort="17459" status="" vrfEncap="VXLAN: 2097153"/>
</imdata>
```

## Viewing ACL Permit and Deny Logs Using the NX-OS CLI

The following steps show how to view ACL log details using the NX-OS-style CLI **show acllog** command.

The syntax for the Layer 3 command is **show acllog {permit | deny} l3 {pkt | flow} tenant <tenant\_name> vrf <vrf\_name> srcip <source\_ip> dstip <destination\_ip> srcport <source\_port> dstport <destination\_port> protocol <protocol> srcintf <source\_interface> start-time <startTime> end-time <endTime> detail**

The syntax for the Layer 2 command is **show acllog {permit | deny} l2 {flow | pkt} tenant <tenant\_name> vrf <VRF\_name> srcintf <source\_interface> vlan <VLAN\_number> detail**



**Note** The full syntax of the **show acllog** command is only available on Generation 2 Cisco Nexus 9000 series switches (with names that end in EX or FX or later, such as N9K-C93180LC-EX) and Cisco APIC Release 3.2 or later. With Generation 1 switches (with names that do not end in EX or FX) or Cisco APIC releases before 3.2, the available syntax is as above.

In Cisco APIC 3.2 and later, additional keywords are added to both versions of the command, with the **detail** keyword: **[dstEpgName <destination\_EPG\_name>| dstmac <destination\_MAC\_address> | dstpctag <destination\_PCTag> | srcEpgName <source\_EPG\_name> | srcmac <source\_MAC\_address> | srcpctag <source\_PCTag>]**

**Step 1**

The following example shows how to use the **show acllog drop l3 flow tenant common vrf default detail** command to display detailed information about Layer 3 deny logs for the common tenant:

**Example:**

```
apic1# show acllog deny l3 flow tenant common vrf default detail
SrcPcTag : 49153
DstPcTag : 32773
SrcEPG : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg6
DstEPG : uni/tn-TSW_Tenant0/ap-tsw0AP0/epg-tsw0ctx0BD0epg5
SrcIp : 16.0.2.10
DstIp : 19.0.2.10
Protocol : udp
SrcPort : 17459
DstPort : 8721
SrcMAC : 00:00:15:00:00:28
DstMAC : 00:00:12:00:00:25
Node : 101
SrcIntf : port-channel5
VrfEncap : VXLAN: 2097153
```

This example shows the output on Generation 2 switches, with Cisco APIC Release 3.2 or later.

**Step 2**

The following example shows how to use the **show acllog deny l2 flow tenant common vrf tsw0connctx0 detail** command to display detailed information about Layer 3 deny logs for the common tenant:

**Example:**

```
apic1# show acllog deny l2 flow tenant common vrf tsw0connctx0 detail
SrcPcTag DstPcTag SrcEPG DstEPG SrcMAC DstMAC Node SrcIntf
vlan

32773 49153 uni/tn-TSW uni/tn-TSW 00:00:11:00:00:11 11:00:32:00:00:33 101 port-
2
 _Tenant0/ap- _Tenant0/ap- channel8
 tsw0AP0/epg- tsw0AP0/epg-
 tsw0ctx0BD0epg5 tsw0ctx0BD0epg6
```

This example shows the output on Generation 2 switches, with Cisco APIC Release 3.2 or later.

**Step 3**

The following example shows how to use the **show acllog permit l3 pkt tenant <tenant name> vrf <vrf name> [detail]** command to display detailed information about the common VRF ACL Layer 3 permit packets that were sent:

```
apic1# show acllog permit l3 pkt tenant common vrf default detail acllog permit l3 packets detail:
srcIp : 10.2.0.19
```

```

dstIp : 10.2.0.16
protocol : udp
srcPort : 13124
dstPort : 4386
srcIntf : port-channel5
vrfEncap : VXLAN: 2097153
pktLen : 112
srcMacAddr : 00:00:15:00:00:28
dstMacAddr : 00:00:12:00:00:25
timeStamp : 2015-03-17T21:31:14.383+00:00

```

This example shows the output on Generation 1 switches, or with Cisco APIC releases before 3.2.

**Step 4** The following example shows how to use the **show acllog permit l2 pkt tenant <tenant name> vrf <vrf name> srcintf <s interface>** command to view information about default VRF Layer 2 packets sent from interface port-channel15:

```

apic1# show acllog permit l2 pkt tenant common vrf default srcintf port-channel5
acllog permit L2 Packets
 Node srcIntf pktLen timeStamp

 port-channel5 1 2015-03-17T21:
 31:14.383+00:00

```

This example shows the output on Generation 1 switches, or with Cisco APIC releases before 3.2.

## Using Atomic Counter Policies for Gathering Statistics

Atomic counter policies enable you to gather statistics about your traffic between a combination of endpoints, endpoint groups, external interfaces, and IP addresses. The information gathered enables you to detect drops and misrouting in the fabric, which enables you to perform quick debugging and to isolate application connectivity issues.

### Atomic Counters

Atomic Counters are useful for troubleshooting connectivity between endpoints, EPGs, or an application within the fabric. A user reporting application may be experiencing slowness, or atomic counters may be needed for monitoring any traffic loss between two endpoints. One capability provided by atomic counters is the ability to place a trouble ticket into a proactive monitoring mode, for example when the problem is intermittent, and not necessarily happening at the time the operator is actively working the ticket.

Atomic counters can help detect packet loss in the fabric and allow the quick isolation of the source of connectivity issues. Atomic counters require NTP to be enabled on the fabric.

Leaf-to-leaf (TEP to TEP) atomic counters can provide the following:

- Counts of drops, admits, and excess packets
- Short-term data collection such as the last 30 seconds, and long-term data collection such as 5 minutes, 15 minutes, or more
- A breakdown of per-spine traffic (available when the number of TEPs, leaf or VPC, is less than 64)
- Ongoing monitoring

Leaf-to-leaf (TEP to TEP) atomic counters are cumulative and cannot be cleared. However, because 30 second atomic counters reset at 30 second intervals, they can be used to isolate intermittent or recurring problems.

Tenant atomic counters can provide the following:

- Application-specific counters for traffic across the fabric, including drops, admits, and excess packets
- Modes include the following:
  - Endpoint to endpoint MAC address, or endpoint to endpoint IP address. Note that a single target endpoint could have multiple IP addresses associated with it.
  - EPG to EPG with optional drill down
  - EPG to endpoint
  - EPG to \* (any)
  - Endpoint to external IP address



---

**Note** Atomic counters track the amount packets of between the two endpoints and use this as a measurement. They do not take into account drops or error counters in a hardware level.

---

Dropped packets are calculated when there are less packets received by the destination than transmitted by the source.

Excess packets are calculated when there are more packets received by the destination than transmitted by the source.

## Atomic Counters Guidelines and Restrictions

- Use of atomic counters is not supported when the endpoints are in different tenants or in different contexts (VRFs) within the same tenant.
- In Cisco APIC release 3.1(2m) and later, if no statistics have been generated on a path in the lifetime of the fabric, no atomic counters are generated for the path. Also, the **Traffic Map** in the **Visualization** tab (**Operations** > **Visualization** in the Cisco APIC GUI) does not show all paths, only the active paths (paths that had traffic at some point in the fabric lifetime).
- In pure Layer 2 configurations where the IP address is not learned (the IP address is 0.0.0.0), endpoint-to-EPG and EPG-to-endpoint atomic counter policies are not supported. In these cases, endpoint-to-endpoint and EPG-to-EPG policies are supported. External policies are virtual routing and forwarding (VRF)-based, requiring learned IP addresses, and are supported.
- When the atomic counter source or destination is an endpoint, the endpoint must be dynamic and not static. Unlike a dynamic endpoint (fv:CEp), a static endpoint (fv:StCEp) does not have a child object (fv:RsCEpToPathEp) that is required by the atomic counter.
- In a transit topology, where leaf switches are not in full mesh with all spine switches, then leaf-to-leaf (TEP to TEP) counters do not work as expected.
- For leaf-to-leaf (TEP to TEP) atomic counters, once the number of tunnels increases the hardware limit, the system changes the mode from trail mode to path mode and the user is no longer presented with per-spine traffic.
- The atomic counter does not count spine proxy traffic.
- Packets dropped before entering the fabric or before being forwarded to a leaf port are ignored by atomic counters.
- Packets that are switched in the hypervisor (same Port Group and Host) are not counted.

- Atomic counters require an active fabric Network Time Protocol (NTP) policy.
- Atomic counters work for IPv6 sources and destinations, but configuring source and destination IP addresses across IPv4 and IPv6 addresses is not allowed.
- An atomic counter policy configured with fvCEp as the source or destination counts only the traffic that is from/to the MAC and IP addresses that are present in the fvCEp managed objects. If the fvCEp managed object has an empty IP address field, then all traffic to/from that MAC address would be counted regardless of the IP address. If the Cisco APIC has learned multiple IP addresses for an fvCEp, then traffic from only the one IP address in the fvCEp managed object itself is counted as previously stated. To configure an atomic counter policy to or from a specific IP address, use the fvIp managed object as the source or destination.
- If there is an fvIp behind an fvCEp, you must add fvIP-based policies and not fvCEp-based policies.
- Endpoint-to-endpoint atomic counter statistics are not reported for Layer 2 bridged traffic with IPv6 headers when the endpoints belong to the same EPG.
- For atomic counters to work for traffic flowing from an EPG or ESG to an L3Out EPG, configure the L3Out EPG with 0/1 and 128/1 to match all prefixes instead of 0/0.
- If your Cisco APIC has the traffic map mode set to "trial" and the Cisco APIC generated the F1545 fault, the only way that you can clear this fault is by setting the traffic map mode to "path." To change the traffic map mode, go to **Operations > Visualization**, click **Settings**, choose **path** for Mode, then click **Submit**. This will give you tunnel stats per port in both ingress and egress.

The trial mode has a greater chance of reaching the maximum scale index of tunnel logical interfaces. This mode consumes more software and hardware resources. A logical interface is the ID that is associated with the tunnel in the hardware.

If you have a single tunnel between a tunnel endpoint (TEP) you specified the trail mode, it will consume more hardware resources as well. For example, if you have 6 fabric ports and a single tunnel, then hardware consumes a number of entries equal to the number of tunnels multiplied by the number of fabric ports.

For software, if the number of logical interfaces allocated is greater than 2048, you will fail to have an entry in the hardware. As a result, you cannot get the stats. In the case of the atomic counter, this issue may show as drops or excesses.

The path mode has only entries for the TEP. For a vPC, two entries will be installed. Therefore, you have a lower chance of reaching to the maximum limit.

## Configuring Atomic Counters

- 
- Step 1** In the menu bar, click **Tenants**.
- Step 2** In the submenu bar, click the desired tenant.
- Step 3** In the **Navigation** pane, expand the tenant and expand **Policies** and then expand **Troubleshoot**.
- Step 4** Under **Troubleshoot**, expand **Atomic Counter Policy** and choose a traffic topology.  
You can measure traffic between a combination of endpoints, endpoint groups, external interfaces, and IP addresses.
- Step 5** Right-click the desired topology and choose **Add topology Policy** to open an **Add Policy** dialog box.
- Step 6** In the **Add Policy** dialog box, perform the following actions:



- a) In the **Name** field, enter a name for the policy.
- b) choose or enter the identifying information for the traffic source.  
The required identifying information differs depending on the type of source (endpoint, endpoint group, external interface, or IP address).
- c) choose or enter the identifying information for the traffic destination.
- d) (Optional) (Optional) In the **Filters** table, click the + icon to specify filtering of the traffic to be counted.  
In the resulting **Create Atomic Counter Filter** dialog box, you can specify filtering by the IP protocol number (TCP=6, for example) and by source and destination IP port numbers.
- e) Click **Submit** to save the atomic counter policy.

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

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

## Enabling Atomic Counters

To enable using atomic counters to detect drops and misrouting in the fabric and enable quick debugging and isolation of application connectivity issues, create one or more tenant atomic counter policies, which can be one of the following types:

- EP\_to\_EP—Endpoint to endpoint (**dbgacEpToEp**)
- EP\_to\_EPG—Endpoint to endpoint group (**dbgacEpToEpg**)
- EP\_to\_Ext—Endpoint to external IP address (**dbgacEpToExt**)
- EPG\_to\_EP—Endpoint group to endpoint(**dbgacEpgToEp**)
- EPG\_to\_EPG—Endpoint group to endpointing group (**dbgacEpgToEpg**)
- EPG\_to\_IP—Endpoint group to IP address (**dbgacEpgToIp**)
- Ext\_to\_EP—External IP address to endpoint (**dbgacExtToEp**)
- IP\_to\_EPG—IP address to endpoint group (**dbgacIpToEpg**)
- Any\_to\_EP—Any to endpoint (**dbgacAnyToEp**)
- EP\_to\_Any—Endpoint to any (**dbgacEpToAny**)

**Step 1** To create an EP\_to\_EP policy using the REST API, use XML such as the following example:

**Example:**

```
<dbgacEpToEp name="EP_to_EP_Policy" ownerTag="" ownerKey=""
dn="uni/tn-Tenant64/acEpToEp-EP_to_EP_Policy" descr="" adminSt="enabled">
<dbgacFilter name="EP_to_EP_Filter" ownerTag="" ownerKey="" descr=""
srcPort="https" prot="tcp" dstPort="https"/>
</dbgacEpToEp>
```

**Step 2** To create an EP\_to\_EPG policy using the REST API, use XML such as the following example:

**Example:**

```
<dbgacEpToEpg name="EP_to_EPG_Pol" ownerTag="" ownerKey=""
dn="uni/tn-Tenant64/epToEpg-EP_to_EPG_Pol" descr="" adminSt="enabled">
<dbgacFilter name="EP_to_EPG_Filter" ownerTag="" ownerKey="" descr=""
srcPort="http" prot="tcp" dstPort="http"/>
<dbgacRsToAbsEpg tDn="uni/tn-Tenant64/ap-VRF64_app_prof/epg-EPG64"/>
</dbgacEpToEpg>
```

## Troubleshooting Using Atomic Counters with the REST API

**Step 1** To get a list of the endpoint-to-endpoint atomic counters deployed within the fabric and the associated details such as dropped packet statistics and packet counts, use the **dbgEpToEpTsIt** class in XML such as the following example:

**Example:**

```
https://apic-ip-address/api/node/class/dbgEpToEpRslt.xml
```

**Step 2** To get a list of external IP-to-endpoint atomic counters and the associated details, use the **dbgacExtToEp** class in XML such as the following example:

**Example:**

```
https://apic-ip-address/api/node/class/dbgExtToEpRslt.xml
```

## Enabling and Viewing Digital Optical Monitoring Statistics

Real-time digital optical monitoring (DOM) data is collected from SFPs, SFP+, and XFPs periodically and compared with warning and alarm threshold table values. The DOM data collected are transceiver transmit bias current, transceiver transmit power, transceiver receive power, and transceiver power supply voltage.

### Enabling Digital Optical Monitoring Using the GUI

Before you can view digital optical monitoring (DOM) statistics about a physical interface, enable DOM on the leaf or spine interface, using a switch policy, associated to a policy group.

To enable DOM using the GUI:

- Step 1** On the menu bar, choose **Fabric > Fabric Policies**.
- Step 2** In the **Navigation** pane, expand **Policies > Monitoring > Fabric Node Controls**.
- Step 3** Expand **Fabric Node Controls** to see a list of existing policies.
- Step 4** In the **Work** pane, click the **ACTIONS** drop-down menu and select **Create Fabric Node Control**. The **Create Fabric Node Control** dialog box appears.
- Step 5** In the **Create Fabric Node Control** dialog box, perform the following actions:
  - a) In the **Name** field, enter a name for the policy.
  - b) Optional. In the **Description** field, enter a description of the policy.
  - c) Put a check in the box next to **Enable DOM**.

- Step 6** Click **Submit** to create the policy.  
Now you can associate this policy to a policy group and a profile, as described in the following steps.
- Step 7** In the **Navigation** pane, expand **Switch Policies > Policy Groups**.
- Step 8** In the **Work** pane, click the **ACTIONS** drop-down menu and select **Create Leaf Switch Policy Group** (for a spine, **Create Spine Switch Policy Group**).  
The **Create Leaf Switch Policy Group** or **Create Spine Switch Policy Group** dialog box appears.
- Step 9** In the dialog box, perform the following actions:
- In the **Name** field, enter a name for the policy group.
  - From the **Node Control Policy** drop-down menu, choose either an existing policy (such as the one you just created) or a new one by selecting **Create Fabric Node Control**.
  - Click **Submit**.
- Step 10** Attach the policy group you created to a switch as follows:
- In the **Navigation** pane, expand **Switch Policies > Profiles**.
  - In the **Work** pane, click the **ACTIONS** drop-down menu and select **Create Leaf Switch Profile** or **Create Spine Switch Profile**, as appropriate.
  - In the dialog box, enter a name for the profile in the **Name** field.
  - Add the name of the switch you want associated with the profile under **Switch Associations**.
  - From the **Blocks** pull-down menu, check the boxes next to the applicable switches.
  - From the **Policy Group** pull-down menu, select the policy group you created earlier.
  - Click **UPDATE**, then click **Submit**.

## Enabling Digital Optical Monitoring Using the REST API

Before you can view digital optical monitoring (DOM) statistics about a physical interface, enable DOM on the interface.

To enable DOM using the REST API:

- Step 1** Create a fabric node control policy (fabricNodeControlPolicy) as in the following example:

```
<fabricNodeControl dn="uni/fabric/nodecontrol-testdom" name="testdom" control="1"
rn="nodecontrol-testdom" status="created" />
```

- Step 2** Associate a fabric node control policy to a policy group as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<fabricLeNodePGrp dn="uni/fabric/funcprof/lenodepgrp-nodegrp2" name="nodegrp2"
rn="lenodepgrp-nodegrp2" status="created,modified" >

 <fabricRsMonInstFabricPol tnMonFabricPolName="default" status="created,modified" />
 <fabricRsNodeCtrl tnFabricNodeControlName="testdom" status="created,modified" />

</fabricLeNodePGrp>
```

- Step 3** Associate a policy group to a switch (in the following example, the switch is 103) as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<fabricLeafP>
 <attributes>
```

```

<dn>uni/fabric/leprof-leafSwitchProfile</dn>
<name>leafSwitchProfile</name>
<rn>leprof-leafSwitchProfile</rn>
<status>created,modified</status>
</attributes>
<children>
<fabricLeafS>
<attributes>
<dn>uni/fabric/leprof-leafSwitchProfile/leaves-test-typrange</dn>
<type>range</type>
<name>test</name>
<rn>leaves-test-typrange</rn>
<status>created,modified</status>
</attributes>
<children>
<fabricNodeBlk>
<attributes>
<dn>uni/fabric/leprof-leafSwitchProfile/leaves-test-typrange/nodeblk-09533c1d228097da</dn>
<from_>103</from_>
<to_>103</to_>
<name>09533c1d228097da</name>
<rn>nodeblk-09533c1d228097da</rn>
<status>created,modified</status>
</attributes>
</fabricNodeBlk>
</children>
</children>
<fabricRsLeNodePGrp>
<attributes>
<tDn>uni/fabric/funcprof/lenodepgrp-nodegrp2</tDn>
<status>created</status>
</attributes>
</fabricRsLeNodePGrp>
</children>
</fabricLeafS>
</children>
</fabricLeafP>

```

## Viewing Digital Optical Monitoring Statistics With the GUI

To view DOM statistics using the GUI:

### Before you begin

You must have previously enabled digital optical monitoring (DOM) statistics for an interface, before you can view the DOM statistics for it.

- Step 1** In the Menu bar, choose **Fabric** and **Inventory**.
- Step 2** In the Navigation pane, expand the Pod and Leaf node where the physical interface you are investigating is located.
- Step 3** Expand **Interfaces**.
- Step 4** Expand **Physical Interfaces**.
- Step 5** Expand the physical interface you are investigating.
- Step 6** Choose **DOM Stats**.

DOM statistics are displayed for the interface.

---

## Troubleshooting Using Digital Optical Monitoring With the REST API

To view DOM statistics using an XML REST API query:

### Before you begin

You must have previously enabled digital optical monitoring (DOM) on an interface, before you can view the DOM statistics for it.

---

The following example shows how to view DOM statistics on a physical interface, eth1/25 on node-104, using a REST API query:

```
GET https://apic-ip-address/api/node/mo/topology/pod-1/node-104/sys/phys-[eth1/25]/phys/domstats.xml?query-target=children&target-subtree-class=ethpmDOMRxPwrStats&subscription=yes
```

The following response is returned:

```
response : {
 "totalCount": "1",
 "subscriptionId": "72057611234705430",
 "imdata": [
 {"ethpmDOMRxPwrStats": {
 "attributes": {
 "alert": "none",
 "childAction": "",
 "dn": "topology/pod-1/node-104/sys/phys[eth1/25]/phys/domstats/rxpower",
 "hiAlarm": "0.158490",
 "hiWarn": "0.079430",
 "loAlarm": "0.001050",
 "loWarn": "0.002630",
 "modTs": "never",
 "status": "",
 "value": "0.139170"}}}}]
```

---

## Viewing and Understanding Health Scores

The APIC uses a policy model to combine data into a health score. Health scores can be aggregated for a variety of areas such as for infrastructure, applications, or services. The health scores enable you to isolate performance issues by drilling down through the network hierarchy to isolate faults to specific managed objects (MOs). You can view network health by viewing the health of an application (by tenant) or by the health of a leaf switch (by pod).

For more information about health scores, faults, and health score calculation see the *Cisco APIC Fundamentals Guide*.

## Health Score Types

The APIC supports the following health score types:

- **System**—Summarizes the health of the entire network.
- **Leaf**—Summarizes the health of leaf switches in the network. Leaf health includes hardware health of the switch including fan tray, power supply, and CPU.
- **Tenant**—Summarizes the health of a tenant and the tenant's applications.

## Filtering by Health Score

You can filter health scores using the following tools:

- **Health Scroll Bar**—You can use the health scroll bar to dictate which objects are visible; lowering the score allows you to see only objects with a degraded health score.
- **Displaying Degraded Health Scores**—To display only the degraded health scores, click the Gear icon and choose **Show only degraded health score**.

## Viewing Tenant Health

To view application health, click **Tenants** > *tenant-name* in the menu bar, then click the tenant name in the **Navigation** pane. The GUI displays a summary of the tenant's health including applications and EPGs. To drill down on the tenant configuration, double-click the health score.

For a health summary, click the **Health** tab in the **Work** pane. This view of the network displays health scores and relationships between MOs in the network so that you can isolate and resolve performance issues. For example, a common sequence of managed objects in the tenant context is **Tenant > Application profile > Application EPG > EPP > Fabric location > EPG to Path Attachment > Network Path Endpoint > Aggregation Interface > Aggregated Interface > Aggregated Member Interface**.

## Viewing Fabric Health

To view fabric health, click **Fabric** in the menu bar. In the **navigation** pane, choose a pod. The GUI displays a summary of the pod health including nodes. To drill down on part of the fabric configuration, double-click the health score.

For a health summary, click the **Health** tab in the **work** pane. This view of the network displays health scores and relationships between MOs in the network so that you can isolate and resolve performance issues. For example, a common sequence of managed objects in the fabric context is **Pod > Leaf > Chassis > Fan tray slot > Line module slot > Line module > Fabric Port > Layer 1 Physical Interface Configuration > Physical Interface Runtime State**.




---

**Note** Fabric issues, such as physical network problems, can impact tenant performance when MOs are directly related.

---

## Viewing MO Health in Visore

To view the health of an MO in Visore, click the **H** icon.

Use the following MOs to display health information:

- health:Inst
- health:NodeInst
- observer:Node
- observer:Pod

For more information about Visore, see the *Cisco Application Centric Infrastructure Fundamentals* guide.

## Debugging Health Scores Using Logs

You can use the following log files to debug health scores on the APIC:

- svc\_ifc\_eventmgr.log
- svc\_ifc\_observer.log

Check the following items when debugging health scores using logs:

- Verify the source of the syslog (fault or event).
- Check whether a syslog policy is configured on the APIC.
- Check whether the syslog policy type and severity is set correctly.
- You can specify a syslog destination of console, file, RemoteDest, or Prof. ForRemoteDest, ensure that the syslog server is running and reachable.

## Viewing Faults

The steps below explain where to view fault information.

---

**Step 1** Go to a faults window:

- System Faults—From the menu bar, click **System > Faults**.
- Tenant Faults—From the menu bar:
  - a. Click **Tenants > tenant-name**.
  - b. From the **Navigation** pane, click the **Tenants** *tenant name*.
  - c. From the **Work** pane, click the **Faults** tab.
- Fabric Faults—From the menu bar:
  - a. Click **Fabric > Inventory**.
  - b. From the **Navigation** pane, click on a **Pod**

- c. From the **Work** pane, click the **Faults** tab.

A list of faults appears in a summary table.

## Step 2 Double-click on a fault.

The fabric and system tables change to display faults that match the fault code of the fault you clicked on.

- a) From the fabric or system faults, double-click on a fault in the summary table to view more information.

The **Fault Properties** dialog appears displaying the following tabs:

- **General**—Displays the following:
  - **Properties**—Contains information found in the summary table
  - **Details**—Contains fault information found in the summary table, the number of occurrences, the change set, and the original, previous, and highest severity level for the chosen fault.
- **Troubleshooting**—Displays the following:
  - **Troubleshooting**—Contains troubleshooting information that includes an explanation of the fault and the recommended action.
  - **Audit log**—A tool that enables you to view the history of user-initiated events before the fault occurred. The history is displayed in a list by a specified number of minutes. You can adjust the number of minutes by clicking the drop-down arrow.
- **History**—Displays history information of the affected object

# Enabling Port Tracking for Uplink Failure Detection

This section explains how to enable port tracking using the GUI, NX-OS CLI, and the REST API.

## Port Tracking Policy for Fabric Port Failure Detection

Fabric port failure detection can be enabled in the port tracking system settings. The port tracking policy monitors the status of fabric ports between leaf switches and spine switches, and ports between tier-1 leaf switches and tier-2 leaf switches. When an enabled port tracking policy is triggered, the leaf switches take down all access interfaces on the switch that have EPGs deployed on them.

If you enabled the **Include APIC ports when port tracking is triggered** option, port tracking disables Cisco Application Policy Infrastructure Controller (APIC) ports when the leaf switch loses connectivity to all fabric ports (that is, there are 0 fabric ports). Enable this feature only if the Cisco APICs are dual- or multihomed to the fabric. Bringing down the Cisco APIC ports helps in switching over to the secondary port in the case of a dual-homed Cisco APIC.




---

**Note** Port tracking is located under **System > System Settings > Port Tracking**.

---



The port tracking policy specifies the number of fabric port connections that trigger the policy, and a delay timer for bringing the leaf switch access ports back up after the number of specified fabric ports is exceeded.

The following example illustrates how a port tracking policy behaves:

- The port tracking policy specifies that the threshold of active fabric port connections each leaf switch that triggers the policy is 2.
- The port tracking policy triggers when the number of active fabric port connections from the leaf switch to the spine switches drops to 2.
- Each leaf switch monitors its fabric port connections and triggers the port tracking policy according to the threshold specified in the policy.
- When the fabric port connections come back up, the leaf switch waits for the delay timer to expire before bringing its access ports back up. This gives the fabric time to reconverge before allowing traffic to resume on leaf switch access ports. Large fabrics may need the delay timer to be set for a longer time.



---

**Note** Use caution when configuring this policy. If the port tracking setting for the number of active spine ports that triggers port tracking is too high, all leaf switch access ports will be brought down.

---

## Configuring Port Tracking Using the GUI

This procedure explains how to use the Port Tracking feature using the GUI.

- 
- Step 1** From the **System** menu, select **System Settings**.
  - Step 2** In the navigation pane, select **Port Tracking**.
  - Step 3** Turn on the Port Tracking feature by selecting **on** next to **Port tracking state**.
  - Step 4** Turn off the Port Tracking feature by selecting **off** next to Port tracking state under Properties.
  - Step 5** (Optional) Reset the **Delay restore timer** from the default (120 seconds).
  - Step 6** Enter the maximum number of active spine links (any configuration value from 0 - 12) that are up before port tracking is triggered.
  - Step 7** Click **Submit** to push your desired Port Tracking configuration to all switches on the fabric.
- 

## Port Tracking Using the NX-OS CLI

This procedure explains how to use the Port Tracking feature using the NX-OS CLI.

- 
- Step 1** Turn on the Port Tracking feature as follows:

**Example:**

```
apicl# show porttrack
Configuration
Admin State : on
```

```
Bringup Delay(s) : 120
Bringdown # Fabric Links up : 0
```

**Step 2** Turn off the Port Tracking feature as follows:

**Example:**

```
apic1# show porttrack
Configuration
Admin State : off
Bringup Delay(s) : 120
Bringdown # Fabric Links up : 0
```

## Port Tracking Using the REST API

### Before you begin

This procedure explains how to use the Port Tracking feature using the REST API.

**Step 1** Turn on the Port Tracking feature using the REST API as follows (**admin state: on**):

```
<polUni>
<infraInfra dn="uni/infra">
<infraPortTrackPol name="default" delay="5" minlinks="4" adminSt="on">

</infraPortTrackPol>
</infraInfra>
</polUni>
```

**Step 2** Turn off the Port Tracking feature using the REST API as follows (**admin state: off**):

```
<polUni>
<infraInfra dn="uni/infra">
<infraPortTrackPol name="default" delay="5" minlinks="4" adminSt="off">

</infraPortTrackPol>
</infraInfra>
</polUni>
```

## Configuring SNMP for Monitoring and Managing Devices

This section explains how to configure SNMP using the GUI.

### About SNMP

The Cisco Application Centric Infrastructure (ACI) provides extensive SNMPv1, v2, and v3 support, including Management Information Bases (MIBs) and notifications (traps). The SNMP standard allows any third-party applications that support the different MIBs to manage and monitor the Cisco ACI fabric.

SNMPv3 provides extended security. Each SNMPv3 device can be selectively enabled or disabled for SNMP service. In addition, each device can be configured with a method of handling SNMPv1 and v2 requests.

Beginning in the 4.2(6) release, SNMPv3 supports the Secure Hash Algorithm-2 (SHA-2) authentication type. For more information about using SNMP, see the *Cisco ACI MIB Quick Reference*.

## SNMP Access Support in Cisco ACI



**Note** For the complete list of MIBs supported in Cisco Application Centric Infrastructure (ACI), see <http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/mib/list/mib-support.html>.

SNMP support in Cisco ACI is as follows:

- SNMP read queries (Get, Next, Bulk, Walk) are supported by leaf and spine switches and by the Cisco Application Policy Infrastructure Controller (APIC).
- SNMP write commands (Set) are not supported by leaf and spine switches or by the Cisco APIC.
- SNMP traps (v1, v2c, and v3) are supported by leaf and spine switches and by the Cisco APIC.



**Note** Cisco ACI supports a maximum of 10 trap receivers.

- SNMPv3 is supported by leaf and spine switches and by the Cisco APIC.
- SNMP using a Cisco APIC IPv6 address is not supported.

**Table 1: SNMP Support Changes by Cisco APIC Release**

Release	Description
1.2(2)	IPv6 support is added for SNMP trap destinations.
1.2(1)	SNMP support for the Cisco APIC controller is added. Previous releases support SNMP only for leaf and spine switches.

## Configuring the SNMP Policy Using the GUI

This procedure configures and enables the SNMP policy on ACI switches.

### Before you begin

To allow SNMP communications, you must configure the following:

- Configure an out-of-band contract allowing SNMP traffic. SNMP traffic typically uses UDP port 161 for SNMP requests.
- Configure the APIC out-of-band IP addresses in the 'mgmt' tenant. Although the out-of-band addresses are configured during APIC setup, the addresses must be explicitly configured in the 'mgmt' tenant before the out-of-band contract will take effect.

- Step 1** In the menu bar, click **Fabric**.
- Step 2** In the submenu bar, click **Fabric Policies**.
- Step 3** In the **Navigation** pane, expand **Pod Policies**.
- Step 4** Under **Pod Policies**, expand **Policies**.
- Step 5** Right-click **SNMP** and choose **Create SNMP Policy**.
- As an alternative to creating a new SNMP policy, you can edit the **default** policy fields in the same manner as described in the following steps.
- Step 6** In the SNMP policy dialog box, perform the following actions:
- In the **Name** field, enter an SNMP policy name.
  - In the **Admin State** field, select **Enabled**.
  - (Optional) In the **SNMP v3 Users** table, click the + icon, enter a **Name**, enter the user's authentication data, and click **Update**.  
This step is needed only if SNMPv3 access is required.
  - In the **Community Policies** table, click the + icon, enter a **Name**, and click **Update**.  
The community policy name can be a maximum of 32 characters in length. The name can contain only letters, numbers and the special characters of underscore (\_), hyphen (-), or period (.). The name cannot contain the @ symbol.
  - In the **Trap Forward Servers** table, click the + icon, enter the **IP Address** of the external server and click **Update**.
- Step 7** Required: To configure allowed SNMP management stations, perform the following actions in the SNMP policy dialog box:
- In the **Client Group Policies** table, click the + icon to open the **Create SNMP Client Group Profile** dialog box.
  - In the **Name** field, enter an SNMP client group profile name.
  - From the **Associated Management EPG** drop-down list, choose the management EPG.
  - In the **Client Entries** table, click the + icon.
  - Enter a client's name in the **Name** field, enter the client's IP address in the **Address** field, and click **Update**.
- Note** When an SNMP management station connects with APIC using SNMPv3, APIC does not enforce the client IP address specified in the SNMP client group profile. For SNMPv3, the management station must exist in the **Client Entries** list, but the IP address need not match, as the SNMPv3 credentials alone are sufficient for access.
- Step 8** Click **OK**.
- Step 9** Click **Submit**.
- Step 10** Under **Pod Policies**, expand **Policy Groups** and choose a policy group or right-click **Policy Groups** and choose **Create POD Policy Group**.
- You can create a new pod policy group or you can use an existing group. The pod policy group can contain other pod policies in addition to the SNMP policy.
- Step 11** In the pod policy group dialog box, perform the following actions:
- In the **Name** field, enter a pod policy group name.
  - From the **SNMP Policy** drop-down list, choose the SNMP policy that you configured and click **Submit**.
- Step 12** Under **Pod Policies**, expand **Profiles** and click **default**.

- Step 13** In the **Work pane**, from the **Fabric Policy Group** drop-down list, choose the pod policy group that you created.
- Step 14** Click **Submit**.
- Step 15** Click **OK**.

---

## Configuring an SNMP Trap Destination Using the GUI

This procedure configures the host information for an SNMP manager that will receive SNMP trap notifications.



---

**Note** ACI supports a maximum of 10 trap receivers. If you configure more than 10, some will not receive notifications.

---

- Step 1** In the menu bar, click **Admin**.
- Step 2** In the submenu bar, click **External Data Collectors**.
- Step 3** In the **Navigation** pane, expand **Monitoring Destinations**.
- Step 4** Right-click **SNMP** and choose **Create SNMP Monitoring Destination Group**.
- Step 5** In the **Create SNMP Monitoring Destination Group** dialog box, perform the following actions:
- In the **Name** field, enter an SNMP destination name and click **Next**.
  - In the **Create Destinations** table, click the + icon to open the **Create SNMP Trap Destination** dialog box.
  - In the **Host Name/IP** field, enter an IPv4 or IPv6 address or a fully qualified domain name for the destination host.
  - Choose the **Port** number and **SNMP Version** for the destination.
  - For SNMP v1 or v2c destinations, enter one of the configured community names as the **Security Name** and choose **noauth** as **v3 Security Level**.  
  
An SNMP v1 or v2c security name can be a maximum of 32 characters in length. The name can contain only letters, numbers and the special characters of underscore (\_), hyphen (-), or period (.). For SNMP v1, the name cannot contain the @ symbol.  
  
For SNMP v2c, in the 4.2(6) release and earlier, the name cannot contain the @ symbol. In the 4.2(7) release and later, the name can contain the @ symbol.
  - For SNMP v3 destinations, enter one of the configured SNMP v3 user names as **Security Name** and choose the desired **v3 Security Level**.  
  
An SNMP v3 security name can be a maximum of 32 characters in length. The name must begin with an uppercase or lowercase letter, and can contain only letters, numbers, and the special characters of underscore (\_), hyphen (-), or period (.). In the 4.2(6) release and earlier, the name cannot contain the @ symbol. In the 4.2(7) release and later, the name can contain the @ symbol.
  - From the **Management EPG** drop-down list, choose the management EPG.
  - Click **OK**.
  - Click **Finish**.
-

## Configuring an SNMP Trap Source Using the GUI

This procedure selects and enables a source object within the fabric to generate SNMP trap notifications.

- 
- Step 1** In the menu bar, click **Fabric**.
- Step 2** In the submenu bar, click **Fabric Policies**.
- Step 3** In the **Navigation** pane, expand **Monitoring Policies**.  
You can create an SNMP source in the **Common Policy**, the **default** policy, or you can create a new monitoring policy.
- Step 4** Expand the desired monitoring policy and choose **Callhome/SNMP/Syslog**.  
If you chose the **Common Policy**, right-click **Common Policy**, choose **Create SNMP Source**, and follow the instructions below for that dialog box.
- Step 5** In the **Work** pane, from the **Monitoring Object** drop-down list, choose **ALL**.
- Step 6** From the **Source Type** drop-down list, choose **SNMP**.
- Step 7** In the table, click the + icon to open the **Create SNMP Source** dialog box.
- Step 8** In the **Create SNMP Source** dialog box, perform the following actions:
- In the **Name** field, enter an SNMP policy name.
  - From the **Dest Group** drop-down list, choose an existing destination for sending notifications or choose **Create SNMP Monitoring Destination Group** to create a new destination.  
The steps for creating an SNMP destination group are described in a separate procedure.
  - Click **Submit**.
- 

## Monitoring the System Using SNMP

You can remotely monitor individual hosts (APIC or another host) and find out the state of any particular node.

You can check the system's CPU and memory usage using SNMP to find out if the CPU is spiking or not. The SNMP, a network management system, uses an SNMP client and accesses information over the APIC and retrieves information back from it.

You can remotely access the system to figure out if the information is in the context of the network management system and you can learn whether or not it is taking too much CPU or memory, or if there are any system or performance issues. Once you learn the source of the issue, you can check the system health and verify whether or not it is using too much memory or CPU.

Refer to the *Cisco ACI MIB Quick Reference Manual* for additional information.

## Configuring SPAN for Traffic Monitoring

This section lists the SPAN guidelines and restrictions and explains how to configure SPAN sessions.

## About SPAN

You can use the Switched Port Analyzer (SPAN) utility to perform detailed troubleshooting or to take a sample of traffic from a particular application host for proactive monitoring and analysis.

SPAN copies traffic from one or more ports, VLANs, or endpoint groups (EPGs) and sends the copied traffic to one or more destinations for analysis by a network analyzer. The process is nondisruptive to any connected devices and is facilitated in the hardware, which prevents any unnecessary CPU load.

You can configure SPAN sessions to monitor traffic received by the source (ingress traffic), traffic transmitted from the source (egress traffic), or both. By default, SPAN monitors all traffic, but you can configure filters to monitor only selected traffic.

You can configure SPAN on a tenant or on a switch. When configured on a switch, you can configure SPAN as a fabric policy or an access policy.

APIC supports the encapsulated remote extension of SPAN (ERSPAN).

Beginning with Release 4.1(1i), the following features are now supported:

- Support for local SPAN with static port-channels as the destination, as long as the sources and the port-channel are local on the same switch.



---

**Note** If you are running APIC release 4.1(1i) or later and you configure a static port-channel as the destination, but then downgrade to a release prior to 4.1(1i), then the SPAN session will go into the administrator disabled state because this feature was not available prior to release 4.1.(1i). There is no other functionality impact.

---

- You no longer have to include the IP prefix of the Layer 3 interface when configuring source SPAN with Layer 3 interface filtering.
- Support for configuring filter groups, which is a grouping of one or more filter entries. Use the filter group to specify the matching criteria that will be used to determine if a received packet should be analyzed using SPAN.
- The SPAN-on-drop feature, which captures packets that are dropped due to forwarding at the ASIC and sends them to a pre-configured SPAN destination. There are 3 types of SPAN-on-drop configuration: access drop using access ports as a SPAN source, fabric drop using fabric ports as a SPAN source, and global drop using all ports on a node as a SPAN source. SPAN-on-drop is configured using regular SPAN (through the CLI, GUI, and REST API) and using troubleshooting SPAN (CLI and REST API, only). For more information about configuring this feature, see *Configuring SPAN Using the GUI*, *Configuring SPAN Using the NX-OS Style CLI*, and *Configuring SPAN Using the REST API*.

## Multinode SPAN

The APIC traffic monitoring policies can span policies at the appropriate places to keep track of all the members of each application group and where they are connected. If any member moves, the APIC automatically pushes the policy to the new leaf. For example, when an endpoint VMotions to a new leaf, the span configuration automatically adjusts.

The ACI fabric supports the following two extensions of encapsulated remote SPAN (ERSPAN) formats:

- Access or tenant SPAN—done for leaf switch front panel ports with or without using VLAN as a filter. The Broadcom Trident 2 ASIC in the leaf switches supports a slightly different version of the ERSPAN Type 1 format. It differs from the ERSPAN Type 1 format defined in the document referenced above in that the GRE header is only 4 bytes and there is no sequence field. The GRE header is always encoded with the following – 0x000088be. Even though 0x88be indicates ERSPAN Type 2, the remaining 2 bytes of the fields identify this as an ERSPAN Type 1 packet with a GRE header of 4 bytes.
- Fabric SPAN—done in leaf switches by the Northstar ASIC or by the Alpine ASIC in the spine switches. While these ASICs support ERSPAN Type 2 and 3 formats, the ACI fabric currently only supports ERSPAN Type 2 for fabric SPAN, as documented in the base-line document referenced above.

Refer to the IETF Internet Draft at the following URL for descriptions of ERSPAN headers: <https://tools.ietf.org/html/draft-foschiano-erspan-00>.

## SPAN Guidelines and Restrictions



- 
- Note** Many guidelines and restrictions depend on whether the switch is a generation 1 or generation 2 switch. The generation of the switch is defined as follows:
- Generation 1 switches are identified by the lack of a suffix, such as "EX", "FX", or "FX2," at the end of the switch name (for example, N9K-9312TX).
  - Generation 2 switches are identified with a suffix, such as "EX", "FX", or "FX2," at the end of the switch name.
- 
- The type of SPAN supported varies:
    - For generation 1 switches, tenant and access SPAN use the encapsulated remote extension of SPAN (ERSPAN) type I (Version 1 option in the Cisco Application Policy Infrastructure Controller (APIC) GUI).
    - For generation 2 switches, tenant and access SPAN use the encapsulated remote extension of SPAN (ERSPAN) type II (Version 2 option in the Cisco APIC GUI).
    - Fabric SPAN uses ERSPAN type II.
  - When configuring ERSPAN session, if the SPAN source contains a destination and interfaces from a spine switch within a GOLF VRF instance, an L3Out prefix is sent to the GOLF router with the wrong BGP next-hop, breaking connectivity from GOLF to that L3Out.
  - A uSeg EPG or ESG cannot be used as a SPAN source EPG because the SPAN source filter is based on the VLAN ID. Thus, even if an endpoint is classified to a uSeg EPG or an ESG, traffic from the endpoint is mirrored if its VLAN is the VLAN of the SPAN source EPG.
  - You cannot specify an l3extLifP Layer 3 subinterface as a SPAN source. You must use the entire port for monitoring traffic from external sources.
  - In local SPAN for FEX interfaces, the FEX interfaces can only be used as SPAN sources, not SPAN destinations.
    - On generation 1 switches, Tx SPAN does not work for any Layer 3 switched traffic.



- On generation 2 switches, Tx SPAN does not work whether traffic is Layer 2 or Layer 3 switched.

There are no limitations for Rx SPAN.

- For SPAN of FEX fabric port-channel (NIF), the member interfaces are supported as SPAN source interfaces on generation 1 leaf switches.



---

**Note** While it is also possible to configure FEX fabric port-channel (NIF) member interfaces as SPAN source interfaces on generation 2 switches, this is not supported for releases prior to Cisco APIC release 4.1.

---

- For information regarding ERSPAN headers, refer to the IETF Internet Draft at this URL: <https://tools.ietf.org/html/draft-foschiano-erspan-00>.
- ERSPAN destination IP addresses must be learned in the fabric as an endpoint.
- SPAN supports IPv6 traffic but the destination IP address for the ERSPAN cannot be an IPv6 address.
- The individual port member of a port channel or a vPC cannot be configured as the source. Use the port channel, vPC, or vPC component as the source in the SPAN session.
- A fault is not raised on the ERSPAN source group when the destination EPG is deleted or unavailable.
- SPAN filters are supported on generation 2 leaf switches only.
- An access SPAN source supports only one of the following filters at a given time:
  - EPG
  - Routed outside (L3Out)
- When deploying the access SPAN source with an L3Out filter, ensure that the L3Out is also deployed on the matching interface:
  - If an L3Out is deployed on a port, a SPAN source must be deployed on the same port.
  - If an L3Out is deployed on a PC, a SPAN source must be deployed on the same PC.
  - If an L3Out is deployed on a vPC, a SPAN source must be deployed on the same vPC.
- An L3Out routed interface and routed sub-interface can be deployed on a port or a PC, but an L3Out SVI can be deployed on a port, PC, or vPC. A SPAN source with an L3Out filter must be deployed accordingly.
- An L3Out filter is not supported in fabric SPAN or tenant SPAN sessions.
- The correct L3Out must be selected in the L3 configuration tab of the EPG bridge domain; otherwise, packet flow for basic L3Out will not work.
- An encapsulation value is mandatory for a routed sub-interface and SVI, but is not applicable for a routed interface. The L3Out sub-interface or SVI encapsulation value must be different from the EPG encapsulation value.
- When an EPG filter is enabled within a SPAN session, ARP packets, which are sent out of the interface in the transit, or tx, direction, will not be spanned.

- SPAN filters are not supported in the following:
  - Fabric ports
  - Fabric and tenant SPAN sessions
  - Spine switches
- L4 port range filter entries will not be added if you attempt to add more L4 port ranges than are officially supported.
- A SPAN session will not come up if you attempt to associate more than the supported filter entries at the SPAN source group level or at the individual SPAN source level.
- Deleted filter entries will remain in TCAM if you add or delete more filters entries than are officially supported.
- See the *Verified Scalability Guide for Cisco ACI* document for SPAN-related limits, such as the maximum number of active SPAN sessions and SPAN filter limitations.
- For the SPAN-on-drop feature, the following guidelines and restrictions apply:
  - The SPAN-on-drop feature is supported on generation 2 leaf switches.
  - The SPAN-on-drop feature only captures packets with forwarding drops in the LUX block, which captures forwarding drop packets at the ingress. The SPAN-on-drop feature cannot capture the BMX (buffer) and RWX (egress) drops.
  - When using the troubleshooting CLI to create a SPAN session with SPAN-on-drop enabled and Cisco APIC as the destination, the session is disabled when 100 MB of data is captured.
  - On a modular chassis, the SPAN-on-drop feature will only work for the packets dropped on the line cards. Packets that are dropped on the fabric card will not be spanned.
  - SPAN-on-drop ACLs with other SPAN ACLs are not merged. If a SPAN-on-drop session is configured on an interface along with ACL-based SPAN, then any packets dropped on that interface will only be sent to the SPAN-on-drop session.
  - You cannot configure SPAN on drop and SPAN ACL on the same session.
  - When an access or fabric port-drop session and a global-drop session are configured, the access or fabric port-drop session takes the priority over the global-drop session.
  - The number of filter entries supported in TCAM =  $(M * S1 * 1 + N * S2 * 2) + (S3 * 2)$ . This is applicable to rx SPAN or tx SPAN, separately. Currently, the maximum filter entries supported in tx or rx SPAN is 480 in each direction when following this formula (and assuming there are no other sources that are configured without filter-group association [means  $S3 = 0$ ] and with 16 port-ranges included). When the number of filter entries exceed the maximum number allowed, a fault will be raised. Note that you can specify Layer 4 port ranges in the filter entry. However, sixteen Layer 4 ports are programmed into the hardware as a single filter entry.

**Note**

- M=The number of IPv4 filters
  - S1=The number of sources with IPv4 filters
  - N=The number of IPv6 filters
  - S2=The number of sources with IPv6 filters
  - S3=The number of sources with no filter group association
- 
- With MAC pinning configured in the LACP policy for a PC or vPC, the PC member ports will be placed in the LACP individual port mode and the PC is operationally non-existent. Hence, a SPAN source configuration with such a PC will fail, resulting in the generation of the "No operational src/dst" fault. With the MAC pinning mode configured, SPAN can be configured only on individual ports.
  - A packet that is received on a Cisco Application Centric Infrastructure (ACI) leaf switch will be spanned only once, even if span sessions are configured on both the ingress and egress interfaces.
  - When you use a routed outside SPAN source filter, you see only unicast in the Tx direction. In the Rx direction, you can see unicast, broadcast, and multicast.
  - An L3Out filter is not supported for transmit multicast SPAN. An L3Out is represented as a combination of sclass/dclass in the ingress ACL filters and can therefore match unicast traffic only. Transmit multicast traffic can be spanned only on ports and port-channels.
  - You can use a port channel interface as a SPAN destination only on -EX and later switches.
  - You cannot configure multiple SPAN sessions with the same source interface when a SPAN filter (5-tuple filter) is applied.
  - The local SPAN destination port of a leaf switch does not expect incoming traffic. You can ensure that the switch drops incoming SPAN destination port traffic by configuring a Layer 2 interface policy and setting the **VLAN Scope** property to **Port Local scope** instead of **Global scope**. Apply this policy to the SPAN destination ports. You can configure a Layer 2 interface policy by going to the following location in the GUI: **Fabric > Access Policies > Policies > Interface > L2 Interface**.
  - When you configure SPAN for a given packet, SPAN is supported for the packet only once. If traffic is selected by SPAN in Rx for the first SSN, the traffic will not be selected by SPAN again in Tx for a second SSN. Thus, when the SPAN session ingress and egress port sits on a single switch, the SPAN session capture will be one-way only. The SPAN session cannot display two-way traffic.
  - A SPAN ACL filter configured in the filter group does not filter the broadcast, unknown-unicast and multicast (BUM) traffic that egresses the access interface. A SPAN ACL in the egress direction works only for unicast IPv4 or IPv6 traffic.
  - When configuring a SPAN destination as a local port, EPGs cannot be deployed to that interface.
  - In a leaf switch, a SPAN source with a VRF filter will match all regular bridge domains and all Layer 3 SVIs under the VRF instance.
  - In a spine switch, a SPAN source with a VRF matches only the configured VRF VNID traffic and a bridge domain filter will match only the bridge domain VNID traffic.

## Configuring SPAN Using the GUI

### Configuring a Tenant SPAN Session Using the Cisco APIC GUI

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

- 
- Step 1** In the menu bar, click **Tenants**.
- Step 2** In the submenu bar, click the tenant that contains the source endpoint.
- Step 3** In the **Navigation** pane, expand the tenant, expand **Policies > Troubleshooting > SPAN**.  
Two nodes appear under **SPAN**: **SPAN Destination Groups** and **SPAN Source Groups**.
- Step 4** From the **Navigation** pane, right-click **SPAN Source Groups** and choose **Create SPAN Source Group**. The **Create SPAN Source Group** dialog appears.
- Step 5** Enter the appropriate values in the required fields of the **Create SPAN Source Group** dialog box.
- Note** For a description of a field, click the information icon (?) at the top-right corner of the dialog box to display the help file.
- Step 6** Expand the **Create Sources** table to open the **Create SPAN Source** dialog box.
- Step 7** Enter the appropriate values in the **Create SPAN Source** dialog box fields.
- Note** For the explanation of a field, click the help icon (?) to view the help file.
- Step 8** When finished creating the SPAN source, click **OK**.  
You return to the **Create SPAN Source Group** dialog box.
- Step 9** When finished entering values in the **Create SPAN Source Group** dialog box fields, click **Submit**.
- 

#### What to do next

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

### Configuring a SPAN Filter Group Using the APIC GUI

- 
- Step 1** In the menu bar, click on **Fabric** and in the submenu bar click on **Access Policies**.
- Step 2** In the **Navigation** pane, expand **Policies > Troubleshooting**, and expand **SPAN**.
- Step 3** Under **SPAN**, right-click **SPAN Filter Groups** and choose **Create SPAN Filter Group**. The **Create Filter Group** dialog appears.
- Step 4** Enter a name for the SPAN filter group, then expand the **Filter Entries** table to enter values into the following fields:

- **Source IP Prefix:** Enter a source IP address in the form of *IP-address/mask*. Both IPv4 and IPv6 addresses are supported. Use a value of **0.0.0.0** to denote an IPv4 address **any** entry in this field or use a value of **::** to denote an IPv6 address **any** entry in this field.
- **First Source Port:** Enter the first source Layer 4 port. This field, together with the **Last Source Port** field, specifies a port range for filtering source ports. Use a value of **0** to denote an **any** entry in this field.
- **Last Source Port:** Enter the last source Layer 4 port. This field, together with the **First Source Port** field, specifies a port range for filtering source ports. Use a value of **0** to denote an **any** entry in this field.
- **Destination IP Prefix:** Enter a destination IP address in the form of *IP-address/mask*. Both IPv4 and IPv6 addresses are supported. Use a value of **0.0.0.0** to denote an IPv4 address **any** entry in this field or use a value of **::** to denote an IPv6 address **any** entry in this field.
- **First Destination Port:** Enter the first destination Layer 4 port. This field, together with the **Last Destination Port** field, specifies a port range for filtering destination ports. Use a value of **0** to denote an **any** entry in this field.
- **Last Destination Port:** Enter the last destination Layer 4 port. This field, together with the **First Destination Port** field, specifies a port range for filtering destination ports. Use a value of **0** to denote an **any** entry in this field.
- **IP Protocol:** Enter the IP protocol. Use a value of **0** to denote an **any** entry in this field.

**Step 5** Click **Update**, then click **Submit** when you have entered the appropriate values into each of the fields in this form.

---

## Configuring an Access SPAN Policy Using the Cisco APIC GUI

This procedure guides you through the Cisco APIC GUI to configure an access SPAN policy. The configuration procedure requires entering values in the fields of one or more GUI dialog boxes. To understand a field and determine a valid value, view the help file by clicking the help icon (?) at the top-right corner of the dialog box.

---

**Step 1** In the menu bar, click on **Fabric > Access Policies**.

**Step 2** In the **Navigation** pane, expand **Policies > Troubleshooting > SPAN**.

Three nodes appear under **SPAN**: **SPAN Source Groups**, **SPAN Filter Groups**, and **SPAN Destination Groups**.

**Step 3** Right-click **SPAN Source Groups** and choose **Create SPAN Source Group**.  
The **Create SPAN Source Group** dialog appears.

**Step 4** Enter the appropriate values in the **Create SPAN Source Group** dialog box fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 5** Expand the **Create Sources** table to open the **Create SPAN Source** dialog box and enter the appropriate values in the required fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 6** In the **Create SPAN Source** dialog box, expand **Add Source Access Paths** to specify the source path.  
The **Associate Source to Path** dialog box appears.

**Step 7** Enter the appropriate values in the **Associate Source to Path** dialog box fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 8** When finished associating the source to a path, click **OK**.

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

**Step 9** When finished configuring the SPAN source, click **OK**.

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

**Step 10** When finished configuring the SPAN source group, click **Submit**.

### What to do next

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

## Configuring a Fabric SPAN Policy Using the Cisco APIC GUI

This section guides you through the Cisco APIC GUI to create a fabric SPAN policy. The configuration procedure requires entering values in the fields of one or more GUI dialog boxes. To understand a field and determine a valid value, view the help file by clicking the help icon (?) at the top-right corner of the dialog box.

**Step 1** In the menu bar, click on **Fabric > Fabric Policies**.

**Step 2** In the **Navigation** pane, expand **Policies > Troubleshooting > SPAN**.

Three nodes appear under **SPAN**: **SPAN Source Groups**, **SPAN Filter Groups**, and **SPAN Destination Groups**.

**Step 3** Right-click **SPAN Source Groups** and choose **Create SPAN Source Group**.  
The **Create SPAN Source Group** dialog appears.

**Step 4** Enter the appropriate values in the **Create SPAN Source Group** dialog box fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 5** Expand the **Create Sources** table to open the **Create SPAN Source** dialog box.

**Step 6** Enter the appropriate values in the **Create SPAN Source** dialog box fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 7** When finished, click **OK**.

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

**Step 8** When finished entering values in the **Create SPAN Source Group** dialog box fields, click **Submit**.

### What to do next

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

## Configuring a Layer 3 EPG SPAN Session for External Access Using the APIC GUI

This procedure shows how to configure a Layer 3 EPG SPAN policy for External Access using the Cisco APIC GUI. The configuration procedure requires entering values in the fields of one or more GUI dialog boxes. To understand a field and determine a valid value, view the help file by clicking the help icon (?) at the top-right corner of the dialog box.

---

**Step 1** In the menu bar, click on **Fabric > Access Policies**.

**Step 2** In the **Navigation** pane, expand **Policies > Troubleshooting > SPAN**.

Three nodes appear under **SPAN**: **SPAN Source Groups**, **SPAN Filter Groups**, and **SPAN Destination Groups**.

**Step 3** Right-click **SPAN Source Groups** and choose **Create SPAN Source Group**.  
The **Create SPAN Source Group** dialog appears.

**Step 4** Enter the appropriate values in the **Create SPAN Source Group** dialog box fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 5** In the **Filter Group** field, select or create a filter group.

See [Configuring a SPAN Filter Group Using the APIC GUI, on page 54](#) for more information.

**Step 6** Expand the **Create Sources** table to open the **Create SPAN Source** dialog box and perform the following actions:

- a) Enter a **Name** for the source policy.
- b) Choose a **Direction** option for the traffic flow.
- c) (Optional) Click to place a check mark in the **Span Drop Packets** check box. When checked, the SPAN-on-drop feature is enabled.

**Note** For more information about the SPAN-on-drop feature, click the help icon (?) to view the help file.

- d) For external access, click **Routed Outside** in the **Type** field.

**Note** If **Routed Outside** is chosen for external access, then the **Name**, **Address**, and **Encap** fields appear to configure the **L3 Outside**.

- e) Expand **Add Source Access Paths** to specify the source path.

The **Associate Source to Path** dialog box appears.

- f) Enter the appropriate values in the **Associate Source to Path** dialog box fields.

**Note** For the explanation of a field, click the help icon (?) to view the help file.

- g) When finished associating the source to a path, click **OK**.

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

- h) When finished configuring the SPAN source, click **OK**.

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

**Step 7** When finished configuring the SPAN source group, click **Submit**.

---

**What to do next**

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

**Configuring a Destination Group for an Access SPAN Policy Using the Cisco APIC GUI**

This section guides you through the Cisco APIC GUI to create a destination group for an access SPAN policy. The configuration procedure requires entering values in the fields of one or more GUI dialog boxes. To understand a field and determine a valid value, view the help file by clicking the help icon (?) at the top-right corner of the dialog box.

Creating a SPAN destination group and source enables you to use a traffic analyzer at the SPAN destination to observe the data packets from the SPAN source and verify the packet format, addresses, protocols, and other information.

- 
- Step 1** In the menu bar, click on **Fabric > Access Policies**.
- Step 2** In the **Navigation** pane, expand **Policies > Troubleshooting > SPAN**.  
Three nodes appear under **SPAN**: **SPAN Source Groups**, **SPAN Filter Groups**, and **SPAN Destination Groups**.
- Step 3** Right-click **SPAN Destination Groups** and choose **Create SPAN Destination Group**.  
The **Create SPAN Destination Group** dialog appears.
- Step 4** Enter the appropriate values in the **Create SPAN Destination Group** dialog box fields.  
**Note** For the explanation of a field, click the help icon (?) to view the help file.
- Step 5** When finished, click **Submit**.  
The destination group is created.
- 

**Configuring a Destination Group for a Fabric SPAN Policy Using the Cisco APIC GUI**

This section guides you through the Cisco APIC GUI to create a destination group for a fabric SPAN policy. The configuration procedure requires entering values in the fields of one or more GUI dialog boxes. To understand a field and determine a valid value, view the help file by clicking the help icon (?) at the top-right corner of the dialog box.

Creating a SPAN destination group and source enables you to use a traffic analyzer at the SPAN destination to observe the data packets from the SPAN source and verify the packet format, addresses, protocols, and other information.

- 
- Step 1** In the menu bar, click on **Fabric > Fabric Policies**.
- Step 2** In the **Navigation** pane, expand **Policies > Troubleshooting > SPAN**.  
Three nodes appear under **SPAN**: **SPAN Source Groups**, **SPAN Filter Groups**, and **SPAN Destination Groups**.
- Step 3** Right-click **SPAN Destination Groups** and choose **Create SPAN Destination Group**.  
The **Create SPAN Destination Group** dialog appears.
- Step 4** Enter the appropriate values in the **Create SPAN Destination Group** dialog box fields.



**Note** For the explanation of a field, click the help icon (?) to view the help file.

**Step 5** When finished, click **Submit**.  
The destination group is created.

#### What to do next

If not already created, configure a source for the fabric SPAN policy.

## Configuring SPAN Using the NX-OS Style CLI

### Configuring Local SPAN in Access Mode

This is the traditional SPAN configuration local to an Access leaf node. Traffic originating from one or more access ports or port-channels can be monitored and sent to a destination port local to the same leaf node.

#### SUMMARY STEPS

1. **configure terminal**
2. **[no] monitor access session** *session-name*
3. **[no] description** *text*
4. **[no] destination interface ethernet** *slot/port leaf node-id*
5. **[no] source interface ethernet** {[*fex/*]*slot/port | port-range*} **leaf** *node-id*
6. **drop enable**
7. **[no] direction** {**rx** | **tx** | **both**}
8. **[no] filter tenant** *tenant-name* **application** *application-name* **epg** *epg-name*
9. **exit**
10. **[no] destination interface port-channel** *port-channel-name-list leaf node-id*
11. **[no] source interface port-channel** *port-channel-name-list leaf node-id* [**fex** *fex-id*]
12. **[no] filter tenant** *tenant-name* **l3out** *L3Out-name* **vlan** *interface-VLAN*
13. **[no] shutdown**

#### DETAILED STEPS

	Command or Action	Purpose
<b>Step 1</b>	<b>configure terminal</b>  <b>Example:</b> <code>apic1# configure terminal</code>	Enters global configuration mode.
<b>Step 2</b>	<b>[no] monitor access session</b> <i>session-name</i>  <b>Example:</b> <code>apic1(config)# monitor access session mySession</code>	Creates an access monitoring session configuration.

	Command or Action	Purpose
<b>Step 3</b>	<b>[no] description</b> <i>text</i> <b>Example:</b> apicl(config-monitor-access)# <b>description</b> "This is my SPAN session"	Adds a description for this access monitoring session. If the text includes spaces, it must be enclosed in single quotes.
<b>Step 4</b>	<b>[no] destination interface ethernet</b> <i>slot/port leaf node-id</i> <b>Example:</b> apicl(config-monitor-access)# <b>destination interface ethernet</b> 1/2 leaf 101	Specifies the destination interface. The destination interface cannot be a FEX port.
<b>Step 5</b>	<b>[no] source interface ethernet</b> {[ <i>fex</i> ] <i>slot/port   port-range</i> } <b>leaf</b> <i>node-id</i> <b>Example:</b> apicl(config-monitor-access)# <b>source interface ethernet</b> 1/2 leaf 101	Specifies the source interface port or port range.
<b>Step 6</b>	<b>drop enable</b> <b>Example:</b> apicl(config-monitor-access-source)# <b>drop enable</b>	Enables the SPAN on drop feature, which captures all packets that are dropped in the ASIC and sends them to a pre-configured SPAN destination.
<b>Step 7</b>	<b>[no] direction</b> { <i>rx</i>   <i>tx</i>   <i>both</i> } <b>Example:</b> apicl(config-monitor-access-source)# <b>direction</b> <i>tx</i>	Specifies direction of traffic to be monitored. The direction can be configured independently for each source port range.
<b>Step 8</b>	<b>[no] filter tenant</b> <i>tenant-name application application-name epg epg-name</i> <b>Example:</b> apicl(config-monitor-access-source)# <b>filter tenant t1 application appl epg epg1</b>	Filters traffic to be monitored. The filter can be configured independently for each source port range.
<b>Step 9</b>	<b>exit</b> <b>Example:</b> apicl(config-monitor-access-source)# <b>exit</b>	Returns to access monitor session configuration mode.
<b>Step 10</b>	<b>[no] destination interface port-channel</b> <i>port-channel-name-list leaf node-id</i> <b>Example:</b> apicl(config-monitor-access)# <b>destination interface port-channel</b> pc1 leaf 101	Specifies the destination interface. The destination interface cannot be a FEX port.  <b>Note</b> Beginning with Release 4.1(1), support is now available for having a static port-channel as the destination interface, as shown in the example command.
<b>Step 11</b>	<b>[no] source interface port-channel</b> <i>port-channel-name-list leaf node-id [fex fex-id]</i> <b>Example:</b>	Specifies the source interface port channel.  (Enters the traffic direction and filter configuration, not shown here.)

	Command or Action	Purpose
	<code>apic1(config-monitor-access)# source interface port-channel pc5 leaf 101</code>	
<b>Step 12</b>	<p><b>[no] filter tenant <i>tenant-name</i> l3out <i>L3Out-name</i> vlan <i>interface-VLAN</i></b></p> <p><b>Example:</b></p> <pre>apic1(config-monitor-access-source)# filter tenant t1 l3out l3out1 vlan 2820</pre>	<p>Filters traffic to be monitored. The filter can be configured independently for each source port range.</p> <p><b>Note</b> Beginning with Release 4.1(1), you no longer have to specify the IP prefix when configuring L3Out interface filtering, as shown in the example.</p>
<b>Step 13</b>	<p><b>[no] shutdown</b></p> <p><b>Example:</b></p> <pre>apic1(config-monitor-access)# no shut</pre>	Disables (or enables) the monitoring session.

### Examples

This example shows how to configure a local access monitoring session.

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-access)# description "This is my SPAN session"
apic1(config-monitor-access)# destination interface ethernet 1/2 leaf 101
apic1(config-monitor-access)# source interface ethernet 1/1 leaf 101

apic1(config-monitor-access)# drop enable

apic1(config-monitor-access-source)# direction tx
apic1(config-monitor-access-source)# filter tenant t1 application appl epg epg1
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access)# no shut
apic1(config-monitor-access)# show run
Command: show running-config monitor access session mySession
Time: Fri Nov 6 23:55:35 2015
monitor access session mySession
 description "This is my SPAN session"
 destination interface eth 1/2 leaf 101
 source interface eth 1/1 leaf 101
 direction tx
 filter tenant t1 application appl epg epg
 exit
exit
```

## Configuring a SPAN Filter Group Using the NX-OS-Style CLI

These procedures describe how to configure a SPAN filter group and filter entries.

### Step 1 configure

Enters global configuration mode.

**Example:**

```
apic1# configure
```

**Step 2** [no] **monitor access filter-group** *filtergroup-name*

Creates an access monitoring filter group configuration.

**Example:**

```
apic1(config)# monitor access filter-group filtergroup1
```

**Step 3** [no] **filter srcaddress** *source-address* **dstaddress** *destination-address* **srcport-from** *source-from-port* **srcport-to** *source-to-port* **dstport-from** *destination-from-port* **dstport-to** *destination-to-port* **ipproto** *IP-protocol*

Configures the filter entries for the filter group, where:

- *source-address* is a source IP address in the form of *IP-address/mask*. Both IPv4 and IPv6 addresses are supported. Use a value of **0.0.0.0** to denote an IPv4 address **any** entry in this field or use a value of **::** to denote an IPv6 address **any** entry in this field.
- *destination-address* is a destination IP address in the form of *IP-address/mask*. Both IPv4 and IPv6 addresses are supported. Use a value of **0.0.0.0** to denote an IPv4 address **any** entry in this field or use a value of **::** to denote an IPv6 address **any** entry in this field.
- *source-from-port* is the first source Layer 4 port. This field, together with the **srcport-to** field, specifies a port range for filtering source ports. Use a value of **0** to denote an **any** entry in this field.
- *source-to-port* is the last source Layer 4 port. This field, together with the **srcport-from** field, specifies a port range for filtering source ports. Use a value of **0** to denote an **any** entry in this field.
- *destination-from-port* is the first destination Layer 4 port. This field, together with the **dstport-to** field, specifies a port range for filtering destination ports. Use a value of **0** to denote an **any** entry in this field.
- *destination-to-port* is the last destination Layer 4 port. This field, together with the **dstport-from** field, specifies a port range for filtering destination ports. Use a value of **0** to denote an **any** entry in this field.
- *IP-protocol* is the IP protocol. Use a value of **0** to denote an **any** entry in this field.

**Example:**

```
apic1(config-monitor-fltgrp)# filter srcaddress 1.1.1.0/24 dstaddress 0.0.0.0 srcport-from 0 srcport-to 0 dstport-from 0 dstport-to 0 ipproto 20
```

**Step 4** **exit**

Returns to access monitor filter group configuration mode.

**Example:**

```
apic1(config-monitor-fltgrp)# exit
```

**Step 5** **exit**

Exits global configuration mode.

**Example:**

```
apic1(config)# exit
```

## Examples

This example shows how to configure a SPAN filter group and filter entries.

```
apic1# configure
apic1(config)# monitor access filter-group filtergroup1
apic1(config-monitor-fltgrp)# filter srcaddress 1.1.1.0/24 dstaddress 0.0.0.0 srcport-from
 0 srcport-to 0 dstport-from 0 dstport-to 0 ipproto 20
apic1(config-monitor-fltgrp)# exit
apic1(config)# exit
```

## Associating a SPAN Filter Group Using the NX-OS-Style CLI

These procedures describe how to associate a filter group to a SPAN source group.

### Step 1 **configure**

Enters global configuration mode.

#### Example:

```
apic1# configure
```

### Step 2 **[no] monitor access session *session-name***

Creates an access monitoring session configuration.

#### Example:

```
apic1(config)# monitor access session session1
```

### Step 3 **filter-group *filtergroup-name***

Associates a filter group.

#### Example:

```
apic1(config-monitor-access)# filter-group filtergroup1
```

### Step 4 **no filter-group**

Disassociates a filter group, if necessary.

#### Example:

```
apic1(config-monitor-access)# no filter-group
```

### Step 5 **[no] source interface ethernet {[*fex*]/*slot/port* | *port-range*} leaf *node-id***

Specifies the source interface port or port range.

#### Example:

```
apic1(config-monitor-access)# source interface ethernet 1/9 leaf 101
```

### Step 6 **filter-group *filtergroup-name***

Associates a filter group to a SPAN source.

#### Example:

```
apicl(config-monitor-access-source) # filter-group filtergroup2
```

**Step 7**     **exit**

Returns to access monitor filter group configuration mode.

**Example:**

```
apicl(config-monitor-access-source) # exit
```

**Step 8**     **no filter-group**

Disassociates the filter group from a SPAN source, if necessary.

**Example:**

```
apicl(config-monitor-access-source) # no filter-group
```

**Step 9**     **exit**

Returns to access monitor filter group configuration mode.

**Example:**

```
apicl(config-monitor-access) # exit
```

**Step 10**   **exit**

Exits global configuration mode.

**Example:**

```
apicl(config) # exit
```

**Examples**

This example shows how to associate a filter group.

```
apicl# configure
apicl(config) # monitor access session session1
apicl(config-monitor-access) # filter-group filtergroup1
apicl(config-monitor-access) # source interface ethernet 1/9 leaf 101
apicl(config-monitor-access-source) # filter-group filtergroup2
apicl(config-monitor-access-source) # exit
apicl(config-monitor-access-source) # no filter-group
apicl(config-monitor-access) # exit
apicl(config) # exit
```

**Configuring ERSPAN in Access Mode**

In the ACI fabric, an access mode ERSPAN configuration can be used for monitoring traffic originating from access ports, port-channels, and vPCs in one or more leaf nodes.

For an ERSPAN session, the destination is always an endpoint group (EPG) which can be deployed anywhere in the fabric. The monitored traffic is forwarded to the destination wherever the EPG is moved.

## SUMMARY STEPS

1. **configure terminal**
2. **[no] monitor access session** *session-name*
3. **[no] description** *text*
4. **[no] destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*
5. **[no] erspan-id** *flow-id*
6. **[no] ip dscp** *dscp-code*
7. **[no] ip ttl** *ttl-value*
8. **[no] mtu** *mtu-value*
9. **exit**
10. **[no] source interface ethernet** {[*fex/*] *slot/port* | *port-range*} **leaf** *node-id*
11. **[no] source interface port-channel** *port-channel-name-list* **leaf** *node-id* [**fex** *fex-id*]
12. **[no] source interface vpc** *vpc-name-list* **leaf** *node-id1* *node-id2* [**fex** *fex-id1* *fex-id2*]
13. **drop enable**
14. **[no] direction** {**rx** | **tx** | **both**}
15. **[no] filter tenant** *tenant-name* **application** *application-name* **epg** *epg-name*
16. **exit**
17. **[no] shutdown**

## DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> <code>apic1# configure terminal</code>	Enters global configuration mode.
Step 2	<b>[no] monitor access session</b> <i>session-name</i>  <b>Example:</b> <code>apic1(config)# monitor access session mySession</code>	Creates an access monitoring session configuration.
Step 3	<b>[no] description</b> <i>text</i>  <b>Example:</b> <code>apic1(config-monitor-access)# description "This is my access ERSPAN session"</code>	Adds a description for this monitoring session. If the text includes spaces, it must be enclosed in single quotes.
Step 4	<b>[no] destination tenant</b> <i>tenant-name</i> <b>application</b> <i>application-name</i> <b>epg</b> <i>epg-name</i> <b>destination-ip</b> <i>dest-ip-address</i> <b>source-ip-prefix</b> <i>src-ip-address</i>  <b>Example:</b> <code>apic1(config-monitor-access)# destination tenant t1 application appl epg epg1 destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1</code>	Specifies the destination interface as a tenant and enters destination configuration mode.
Step 5	<b>[no] erspan-id</b> <i>flow-id</i>  <b>Example:</b>	Configures the ERSPAN ID for the ERSPAN session. The ERSPAN range is from 1 to 1023.

	Command or Action	Purpose
	<code>apic1(config-monitor-access-dest)# erspan-id 100</code>	
<b>Step 6</b>	<b>[no] ip dscp dscp-code</b> <b>Example:</b> <code>apic1(config-monitor-access-dest)# ip dscp 42</code>	Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 64.
<b>Step 7</b>	<b>[no] ip ttl ttl-value</b> <b>Example:</b> <code>apic1(config-monitor-access-dest)# ip ttl 16</code>	Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
<b>Step 8</b>	<b>[no] mtu mtu-value</b> <b>Example:</b> <code>apic1(config-monitor-access-dest)# mtu 9216</code>	Configures the maximum transmit unit (MTU) size for the ERSPAN session. The range is 64 to 9216 bytes.
<b>Step 9</b>	<b>exit</b> <b>Example:</b> <code>apic1(config-monitor-access-dest)#</code>	Returns to monitor access configuration mode.
<b>Step 10</b>	<b>[no] source interface ethernet {[fex/] slot/port   port-range} leaf node-id</b> <b>Example:</b> <code>apic1(config-monitor-access)# source interface eth 1/2 leaf 101</code>	Specifies the source interface port or port range.
<b>Step 11</b>	<b>[no] source interface port-channel port-channel-name-list leaf node-id [fex fex-id]</b> <b>Example:</b> <code>apic1(config-monitor-access)# source interface port-channel pc1 leaf 101</code>	Specifies the source interface port-channel.
<b>Step 12</b>	<b>[no] source interface vpc vpc-name-list leaf node-id1 node-id2 [fex fex-id1 fex-id2]</b> <b>Example:</b> <code>apic1(config-monitor-access)# source interface vpc pc1 leaf 101 102</code>	Specifies the source interface vPC.
<b>Step 13</b>	<b>drop enable</b> <b>Example:</b> <code>apic1(config-monitor-access-source)# drop enable</code>	Enables the SPAN on drop feature, which captures all packets that are dropped in the ASIC and sends them to a pre-configured SPAN destination.
<b>Step 14</b>	<b>[no] direction {rx   tx   both}</b> <b>Example:</b> <code>apic1(config-monitor-access-source)# direction tx</code>	Specifies direction of traffic to be monitored. The direction can be configured independently for each source port range.



	Command or Action	Purpose
Step 15	<p><b>[no] filter tenant <i>tenant-name</i> application <i>application-name</i> epg <i>epg-name</i></b></p> <p><b>Example:</b></p> <pre>apic1(config-monitor-access-source)# filter tenant t1 application appl epg epg1</pre>	Filters traffic to be monitored. The filter can be configured independently for each source port range.
Step 16	<p><b>exit</b></p> <p><b>Example:</b></p> <pre>apic1(config-monitor-access-source)# exit</pre>	Returns to access monitor session configuration mode.
Step 17	<p><b>[no] shutdown</b></p> <p><b>Example:</b></p> <pre>apic1(config-monitor-access)# no shut</pre>	Disables (or enables) the monitoring session.

## Examples

This example shows how to configure an ERSPAN access monitoring session.

```
apic1# configure terminal
apic1(config)# monitor access session mySession
apic1(config-monitor-access)# description "This is my access ERSPAN session"
apic1(config-monitor-access)# destination tenant t1 application appl epg epg1 destination-ip
192.0.20.123 source-ip-prefix 10.0.20.1
apic1(config-monitor-access-dest)# erspan-id 100
apic1(config-monitor-access-dest)# ip dscp 42
apic1(config-monitor-access-dest)# ip ttl 16
apic1(config-monitor-access-dest)# mtu 9216
apic1(config-monitor-access-dest)# exit
apic1(config-monitor-access)# source interface eth 1/1 leaf 101
apic1(config-monitor-access-source)# direction tx
apic1(config-monitor-access-source)#drop enable

apic1(config-monitor-access-source)# filter tenant t1 application appl epg epg1
apic1(config-monitor-access-source)# exit
apic1(config-monitor-access)# no shut
apic1(config-monitor-access)# show run
Command: show running-config monitor access session mySession
Time: Fri Nov 6 23:55:35 2015
monitor access session mySession
description "This is my ERSPAN session"
source interface eth 1/1 leaf 101
direction tx
filter tenant t1 application appl epg epg1
exit
destination tenant t1 application appl epg epg1 destination-ip 192.0.20.123
source-ip-prefix 10.0.20.1
ip dscp 42
ip ttl 16
erspan-id 9216
mtu 9216
exit
exit
```

This example shows how to configure a port-channel as a monitoring source.

```
apicl(config-monitor-access)# source interface port-channel pc3 leaf 105
```

This example shows how to configure a one leg of a vPC as a monitoring source.

```
apicl(config-monitor-access)# source interface port-channel vpc3 leaf 105
```

This example shows how to configure a range of ports from FEX 101 as a monitoring source.

```
apicl(config-monitor-access)# source interface eth 101/1/1-2 leaf 105
```

## Configuring ERSPAN in Fabric Mode

In the ACI fabric, a fabric mode ERSPAN configuration can be used for monitoring traffic originating from one or more fabric ports in leaf or spine nodes. Local SPAN is not supported in fabric mode.

For an ERSPAN session, the destination is always an endpoint group (EPG) which can be deployed anywhere in the fabric. The monitored traffic is forwarded to the destination wherever the EPG is moved. In the fabric mode, only fabric ports are allowed as source, but both leaf and spine switches are allowed.

### SUMMARY STEPS

1. **configure terminal**
2. [no] **monitor fabric session** *session-name*
3. [no] **description** *text*
4. [no] **destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*
5. [no] **erspan-id** *flow-id*
6. [no] **ip dscp** *dscp-code*
7. [no] **ip ttl** *ttl-value*
8. [no] **mtu** *mtu-value*
9. **exit**
10. [no] **source interface ethernet** {*slot/port* | *port-range*} **switch** *node-id*
11. **drop enable**
12. [no] **direction** {**rx** | **tx** | **both**}
13. [no] **filter tenant** *tenant-name* **bd** *bd-name*
14. [no] **filter tenant** *tenant-name* **vrf** *vrf-name*
15. **exit**
16. [no] **shutdown**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	configure terminal Example:	Enters global configuration mode.

	Command or Action	Purpose
	<code>apic1# configure terminal</code>	
<b>Step 2</b>	<b>[no] monitor fabric session</b> <i>session-name</i> <b>Example:</b> <code>apic1(config)# monitor fabric session mySession</code>	Creates a fabric monitoring session configuration.
<b>Step 3</b>	<b>[no] description</b> <i>text</i> <b>Example:</b> <code>apic1(config-monitor-fabric)# description "This is my fabric ERSPAN session"</code>	Adds a description for this monitoring session. If the text includes spaces, it must be enclosed in single quotes.
<b>Step 4</b>	<b>[no] destination tenant</b> <i>tenant-name</i> <b>application</b> <i>application-name</i> <b>epg</b> <i>epg-name</i> <b>destination-ip</b> <i>dest-ip-address</i> <b>source-ip-prefix</b> <i>src-ip-address</i> <b>Example:</b> <code>apic1(config-monitor-fabric)# destination tenant t1 application appl epg epg1 destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1</code>	Specifies the destination interface as a tenant and enters destination configuration mode.
<b>Step 5</b>	<b>[no] erspan-id</b> <i>flow-id</i> <b>Example:</b> <code>apic1(config-monitor-fabric-dest)# erspan-id 100</code>	Configures the ERSPAN ID for the ERSPAN session. The ERSPAN range is from 1 to 1023.
<b>Step 6</b>	<b>[no] ip dscp</b> <i>dscp-code</i> <b>Example:</b> <code>apic1(config-monitor-fabric-dest)# ip dscp 42</code>	Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 64.
<b>Step 7</b>	<b>[no] ip ttl</b> <i>tll-value</i> <b>Example:</b> <code>apic1(config-monitor-fabric-dest)# ip ttl 16</code>	Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
<b>Step 8</b>	<b>[no] mtu</b> <i>mtu-value</i> <b>Example:</b> <code>apic1(config-monitor-fabric-dest)# mtu 9216</code>	Configures the maximum transmit unit (MTU) size for the ERSPAN session. The range is 64 to 9216 bytes.
<b>Step 9</b>	<b>exit</b> <b>Example:</b> <code>apic1(config-monitor-fabric-dest)#</code>	Returns to monitor access configuration mode.
<b>Step 10</b>	<b>[no] source interface ethernet</b> <i>{slot/port   port-range}</i> <b>switch</b> <i>node-id</i> <b>Example:</b> <code>apic1(config-monitor-fabric)# source interface eth 1/2 switch 101</code>	Specifies the source interface port or port range.

	Command or Action	Purpose
<b>Step 11</b>	<b>drop enable</b> <b>Example:</b> apicl(config-monitor-fabric-source) # <b>drop enable</b>	Enables the SPAN on drop feature, which captures all packets that are dropped in the ASIC and sends them to a pre-configured SPAN destination.
<b>Step 12</b>	<b>[no] direction {rx   tx   both}</b> <b>Example:</b> apicl(config-monitor-fabric-source) # <b>direction tx</b>	Specifies direction of traffic to be monitored. The direction can be configured independently for each source port range.
<b>Step 13</b>	<b>[no] filter tenant <i>tenant-name</i> bd <i>bd-name</i></b> <b>Example:</b> apicl(config-monitor-fabric-source) # <b>filter tenant t1 bd bd1</b>	Filters traffic by bridge domain.
<b>Step 14</b>	<b>[no] filter tenant <i>tenant-name</i> vrf <i>vrf-name</i></b> <b>Example:</b> apicl(config-monitor-fabric-source) # <b>filter tenant t1 vrf vrf1</b>	Filters traffic by VRF.
<b>Step 15</b>	<b>exit</b> <b>Example:</b> apicl(config-monitor-fabric-source) # <b>exit</b>	Returns to access monitor session configuration mode.
<b>Step 16</b>	<b>[no] shutdown</b> <b>Example:</b> apicl(config-monitor-fabric) # <b>no shut</b>	Disables (or enables) the monitoring session.

## Examples

This example shows how to configure an ERSPAN fabric monitoring session.

```

apicl# configure terminal
apicl(config)# monitor fabric session mySession
apicl(config-monitor-fabric) # description "This is my fabric ERSPAN session"
apicl(config-monitor-fabric) # destination tenant t1 application appl1 epg epg1 destination-ip
192.0.20.123 source-ip-prefix 10.0.20.1
apicl(config-monitor-fabric-dest) # erspan-id 100
apicl(config-monitor-fabric-dest) # ip dscp 42
apicl(config-monitor-fabric-dest) # ip ttl 16
apicl(config-monitor-fabric-dest) # mtu 9216
apicl(config-monitor-fabric-dest) # exit
apicl(config-monitor-fabric) # source interface eth 1/1 switch 101

apicl(config-monitor-fabric-source) # drop enable

apicl(config-monitor-fabric-source) # direction tx
apicl(config-monitor-fabric-source) # filter tenant t1 bd bd1
apicl(config-monitor-fabric-source) # filter tenant t1 vrf vrf1
apicl(config-monitor-fabric-source) # exit

```

```
apic1(config-monitor-fabric)# no shut
```

## Configuring ERSPAN in Tenant Mode

In the ACI fabric, a tenant mode ERSPAN configuration can be used for monitoring traffic originating from endpoint groups within a tenant.

In the tenant mode, traffic originating from a source EPG is sent to a destination EPG within the same tenant. The monitoring of traffic is not impacted if the source or destination EPG is moved within the fabric.

### SUMMARY STEPS

1. **configure terminal**
2. **[no] monitor tenant** *tenant-name* **session** *session-name*
3. **[no] description** *text*
4. **[no] destination tenant** *tenant-name* **application** *application-name* **epg** *epg-name* **destination-ip** *dest-ip-address* **source-ip-prefix** *src-ip-address*
5. **[no] erspan-id** *flow-id*
6. **[no] ip dscp** *dscp-code*
7. **[no] ip ttl** *ttl-value*
8. **[no] mtu** *mtu-value*
9. **exit**
10. **[no] source application** *application-name* **epg** *epg-name*
11. **[no] direction** {*rx* | *tx* | **both**}
12. **exit**
13. **[no] shutdown**

### DETAILED STEPS

	Command or Action	Purpose
Step 1	<b>configure terminal</b>  <b>Example:</b> <pre>apic1# configure terminal</pre>	Enters global configuration mode.
Step 2	<b>[no] monitor tenant</b> <i>tenant-name</i> <b>session</b> <i>session-name</i>  <b>Example:</b> <pre>apic1(config)# monitor tenant session mySession</pre>	Creates a tenant monitoring session configuration.
Step 3	<b>[no] description</b> <i>text</i>  <b>Example:</b> <pre>apic1(config-monitor-tenant)# description "This is my tenant ERSPAN session"</pre>	Adds a description for this access monitoring session. If the text includes spaces, it must be enclosed in single quotes.
Step 4	<b>[no] destination tenant</b> <i>tenant-name</i> <b>application</b> <i>application-name</i> <b>epg</b> <i>epg-name</i> <b>destination-ip</b> <i>dest-ip-address</i> <b>source-ip-prefix</b> <i>src-ip-address</i>  <b>Example:</b>	Specifies the destination interface as a tenant and enters destination configuration mode.

	Command or Action	Purpose
	<code>apic1(config-monitor-tenant)# destination tenant t1 application appl epg epg1 destination-ip 192.0.20.123 source-ip-prefix 10.0.20.1</code>	
<b>Step 5</b>	<b>[no] erspan-id</b> <i>flow-id</i> <b>Example:</b> <code>apic1(config-monitor-tenant-dest)# erspan-id 100</code>	Configures the ERSPAN ID for the ERSPAN session. The ERSPAN range is from 1 to 1023.
<b>Step 6</b>	<b>[no] ip dscp</b> <i>dscp-code</i> <b>Example:</b> <code>apic1(config-monitor-tenant-dest)# ip dscp 42</code>	Configures the differentiated services code point (DSCP) value of the packets in the ERSPAN traffic. The range is from 0 to 64.
<b>Step 7</b>	<b>[no] ip ttl</b> <i>tll-value</i> <b>Example:</b> <code>apic1(config-monitor-tenant-dest)# ip ttl 16</code>	Configures the IP time-to-live (TTL) value for the ERSPAN traffic. The range is from 1 to 255.
<b>Step 8</b>	<b>[no] mtu</b> <i>mtu-value</i> <b>Example:</b> <code>apic1(config-monitor-tenant-dest)# mtu 9216</code>	Configures the maximum transmit unit (MTU) size for the ERSPAN session. The range is 64 to 9216 bytes.
<b>Step 9</b>	<b>exit</b> <b>Example:</b> <code>apic1(config-monitor-tenant-dest)#</code>	Returns to monitor access configuration mode.
<b>Step 10</b>	<b>[no] source application</b> <i>application-name</i> <b>epg</b> <i>epg-name</i> <b>Example:</b> <code>apic1(config-monitor-tenant)# source application app2 epg epg5</code>	Specifies the source interface port or port range.
<b>Step 11</b>	<b>[no] direction</b> { <i>rx</i>   <i>tx</i>   <i>both</i> } <b>Example:</b> <code>apic1(config-monitor-tenant-source)# direction tx</code>	Specifies direction of traffic to be monitored. The direction can be configured independently for each source port range.
<b>Step 12</b>	<b>exit</b> <b>Example:</b> <code>apic1(config-monitor-tenant-source)# exit</code>	Returns to access monitor session configuration mode.
<b>Step 13</b>	<b>[no] shutdown</b> <b>Example:</b> <code>apic1(config-monitor-tenant)# no shut</code>	Disables (or enables) the monitoring session.

### Examples

This example shows how to configure an ERSPAN tenant monitoring session.

```

apicl# configure terminal
apicl(config)# monitor access session mySession
apicl(config-monitor-tenant)# description "This is my tenant ERSPAN session"
apicl(config-monitor-tenant)# destination tenant t1 application appl epg epg1 destination-ip
 192.0.20.123 source-ip-prefix 10.0.20.1
apicl(config-monitor-tenant-dest)# erspan-id 100
apicl(config-monitor-tenant-dest)# ip dscp 42
apicl(config-monitor-tenant-dest)# ip ttl 16
apicl(config-monitor-tenant-dest)# mtu 9216
apicl(config-monitor-tenant-dest)# exit
apicl(config-monitor-tenant)# source application app2 epg epg5
apicl(config-monitor-tenant-source)# direction tx
apicl(config-monitor-tenant-source)# exit
apicl(config-monitor-tenant)# no shut

```

## Configuring a Global SPAN-On-Drop Session Using the NX-OS-Style CLI

This section demonstrates how to create a global drop with all ports on a node as the SPAN source.

### Step 1 **configure terminal**

Enters global configuration mode.

**Example:**

```
apicl# configure terminal
```

### Step 2 **[no] monitor fabric session *session-name***

Creates a fabric monitoring session configuration.

**Example:**

```
apicl(config)# monitor fabric session Spine301-GD-SOD
```

### Step 3 **[no] description *text***

Adds a description for this monitoring session. If the text includes spaces, it must be enclosed in single quotes.

**Example:**

```
apicl(config-monitor-fabric)# description "This is my fabric ERSPAN session"
```

### Step 4 **source global-drop switch**

Enables the SPAN on drop feature, which captures all packets that are dropped in the ASIC and sends them to a pre-configured SPAN destination.

**Example:**

```
apicl(config-monitor-fabric)# source global-drop switch
```

### Step 5 **[no] destination tenant *tenant-name* application *application-name* epg *epg-name* destination-ip *dest-ip-address* source-ip-prefix *src-ip-address***

Specifies the destination interface as a tenant and enters destination configuration mode.

**Example:**

```
apic1(config-monitor-fabric-dest)# destination tenant ERSPAN application A1 epg E1 destination-ip
165.10.10.155 source-ip-prefix 22.22.22.22
```

## Examples

This example shows how to configure a global SPAN-on-Drop session.

```
apic1# configure terminal
apic1(config)# monitor fabric session Spine301-GD-SOD
apic1(config-monitor-fabric)# source global-drop switch
apic1(config-monitor-fabric)# destination tenant ERSPAN application A1 epg E1 destination-ip
179.10.10.179 source-ip-prefix 31.31.31.31
```

# Configuring SPAN Using the REST API

## Configuring a Fabric Destination Group for an ERSPAN Destination Using the REST API

This section demonstrates how to use the REST API to configure a fabric destination group for an ERSPAN destination using the REST API. For a complete list of properties, see the *APIC Management Information Model Reference*.

Configure a fabric destination group for an ERSPAN destination:

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag="">
 <spanDest annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag="">
 <spanRsDestEpg annotation="" dscp="unspecified" finalIp="0.0.0.0" flowId="1" ip="179.10.10.179"
 mtu="1518"srcIpPrefix="20.20.20.2" tDn="uni/tn-ERSPAN/ap-A1/epg-E1" ttl="64" ver="ver2"
 verEnforced="no"/>
 </spanDest>
 </spanDestGrp>
```

## Configuring a Global Drop Source Group Using the REST API

This section demonstrates how to use the REST API to configure a global drop source group using the REST API. For a complete list of properties, see the *APIC Management Information Model Reference*.

Configure a global drop source group:

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Spine-402-GD-SOD" nameAlias="">
 <spanSrc annotation="" descr="" dir="both" name="402" nameAlias="" spanOnDrop="yes">
 <spanRsSrcToNode annotation="" tDn="topology/pod-1/node-402"/>
 </spanSrc>
</spanSrcGrp>
```



```
</spanSrc><spanSpanLbl annotation="" descr="" name="402-dst-179" nameAlias="" tag="yellow-green"/>
</spanSrcGrp>
```

## Configuring a Leaf Port as a SPAN Destination Using the REST API

This section demonstrates how to use the REST API to configure a leaf port as a SPAN destination using the REST API. For a complete list of properties, see the *APIC Management Information Model Reference*.

Configure a leaf port as a SPAN destination:

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest4" nameAlias="" ownerKey="" ownerTag="">
 <spanDest annotation="" descr="" name="Dest4" nameAlias="" ownerKey="" ownerTag="">
 <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-301/pathep-[eth1/18]"/>
 </spanDest>
</spanDestGrp>
```

## Configuring a SPAN Access Source Group Using the REST API

This section demonstrates how to use the REST API to configure a SPAN access source group using the REST API. For a complete list of properties, see the *APIC Management Information Model Reference*.

Configure a SPAN access source group:

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Test-Src2" nameAlias="" ownerKey=""
ownerTag="">
 <spanSrc annotation="" descr="" dir="both" name="Src1" nameAlias="" ownerKey="" ownerTag=""
spanOnDrop="yes">
 <spanRsSrcToPathEp annotation="" tDn="topology/pod-1/paths-301/pathep-[eth1/1]"/>
 </spanSrc>
 <spanSpanLbl annotation="" descr="" name="Dest1" nameAlias="" ownerKey="" ownerTag=""
tag="yellow-green"/>
</spanSrcGrp>
```

## Configuring a SPAN Fabric Source Group Using the REST API

This section demonstrates how to use the REST API to configure a SPAN fabric source group using the REST API. For a complete list of properties, see the *APIC Management Information Model Reference*.

Configure a SPAN fabric source group:

```
POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanSrcGrp adminSt="enabled" annotation="" descr="" name="Test-Src2" nameAlias="" ownerKey=""
ownerTag="">
 <spanSrc annotation="" descr="" dir="both" name="Src1" nameAlias="" ownerKey=""
```

```

 ownerTag="" spanOnDrop="yes">
<spanRsSrcToPathEp annotation="" tDn="topology/pod-1/paths-301/pathep-[eth1/51]"/>
</spanSrc>
<spanSpanLbl annotation="" descr="" name="Dest2" nameAlias="" ownerKey="" ownerTag=""
 tag="yellow-green"/>
</spanSrcGrp>

```

---

## Configuring an Access Destination Group for an ERSPAN Destination Using the REST API

This section demonstrates how to use the REST API to configure an access destination group for an ERSPAN destination using the REST API. For a complete list of properties, see the *APIC Management Information Model Reference*.

Configure an access destination group for an ERSPAN destination.

```

POST https://<APIC_IP>/api/node/mo/uni/infra.xml
<spanDestGrp annotation="" descr="" name="Dest4" nameAlias="" ownerKey=""
 ownerTag="">
 <spanDest annotation="" descr="" name="Dest4" nameAlias="" ownerKey=""
 ownerTag="">
 <spanRsDestPathEp annotation="" mtu="1518" tDn="topology/pod-1/paths-301/pathep-
 [eth1/18]"/>
 </spanDest>
 </spanDestGrp>

```

---

## Using Statistics

Statistics provide real-time measures of observed object and enable trend analysis and troubleshooting. Statistics gathering can be configured for ongoing or on-demand collection and can be collected in cumulative counters and gauges.

Policies define what statistics are gathered, at what intervals, and what actions to take. For example, a policy could raise a fault on an EPG if a threshold of dropped packets on an ingress VLAN is greater than 1000 per second.

Statistics data are gathered from a variety of sources, including interfaces, VLANs, EPGs, application profiles, ACL rules, tenants, or internal APIC processes. Statistics accumulate data in 5-minute, 15-minute, 1-hour, 1-day, 1-week, 1-month, 1-quarter, or 1-year sampling intervals. Shorter duration intervals feed longer intervals. A variety of statistics properties are available, including last value, cumulative, periodic, rate of change, trend, maximum, min, average. Collection and retention times are configurable. Policies can specify if the statistics are to be gathered from the current state of the system or to be accumulated historically or both. For example, a policy could specify that historical statistics be gathered for 5-minute intervals over a period of 1 hour. The 1 hour is a moving window. Once an hour has elapsed, the incoming 5 minutes of statistics are added, and the earliest 5 minutes of data are abandoned.



---

**Note** The maximum number of 5-minute granularity sample records is limited to 12 samples (one hour of statistics). All other sample intervals are limited to 1,000 sample records. For example, hourly granularity statistics can be maintained for up to 41 days.

---

## Viewing Statistics in the GUI

You can view statistics for many objects using the APIC GUI, including application profiles, physical interfaces, bridge domains, and fabric nodes. To view statistics in the GUI, choose the object in the **navigation** pane and click the **STATS** tab.

Follow these steps to view statistics for an interface:

- 
- Step 1** On the menu bar, choose **FABRIC > INVENTORY**.
  - Step 2** In the **Navigation** pane, choose a pod.
  - Step 3** Expand the pod, and expand a switch.
  - Step 4** In the **Navigation** pane, expand **Interfaces** and choose **eth1/1**.
  - Step 5** In the **Work** pane, choose the **STATS** tab.
- 

The APIC displays interface statistics.

### Example

#### What to do next

You can use the following icons in the **Work** pane to manage how the APIC displays statistics:

- Refresh—Manually refreshes statistics.
- Show Table View—Toggles between table and chart views.
- Start or Stop Stats—Enables or disables automatic refresh for statistics.
- Select Stats—Specifies the counters and sample interval to display.
- Download Object as XML—Downloads the object in XML format.
- Measurement Type (Gear icon)—Specifies the statistics measurement type. Options include cumulative, periodic, average, or trend.

## Switch Statistics Commands

You can use the following commands to display statistics on ACI leaf switches.

Command	Purpose
Legacy Cisco Nexus <b>show/clear</b> commands	For more information, see <i>Cisco Nexus 9000 Series NX-OS Configuration Guides</i> .
<b>show platform internal counters port</b> [ <i>port_num</i>   <b>detail</b>   <b>nz</b>   { <b>internal</b> [ <b>nz</b>   <i>int_port_num</i> ]}]	<p>Displays spine port statistics</p> <ul style="list-style-type: none"> <li>• <i>port_num</i>—Front port number without the slot.</li> <li>• <b>detail</b>—Returns SNMP, class and forwarding statistics.</li> <li>• <b>nz</b>—Displays only non-zero values.</li> <li>• <b>internal</b>—Displays internal port statistics.</li> <li>• <i>int_port_num</i>—Internal logical port number. For example, for BCM-0/97, enter 97.</li> </ul> <p><b>Note</b> If there is a link reset, the counters will be zeroed out on the switch. The conditions of counter reset include the following:</p> <ul style="list-style-type: none"> <li>• accidental link reset</li> <li>• manually enabled port (after port is disabled)</li> </ul>
<b>show platform internal counters vlan</b> [ <i>hw_vlan_id</i> ]	Displays VLAN statistics.
<b>show platform internal counters tep</b> [ <i>tunnel_id</i> ]	Displays TEP statistics.
<b>show platform internal counters flow</b> [ <i>rule_id</i>   { <b>dump</b> [ <i>asic_inst</i> ]   [ <b>slice direction</b>   <b>index hw_index</b> ]}]	Displays flow statistics.
<b>clear platform internal counters port</b> [ <i>port_num</i>   { <b>internal</b> [ <i>int_port_num</i> ]}]	Clears port statistics.
<b>clear platform internal counters vlan</b> [ <i>hw_vlan_id</i> ]	Clears VLAN counters.
<b>debug platform internal stats logging level</b> <i>log_level</i>	Sets the debug logging level.
<b>debug platform internal stats logging</b> { <b>err</b>   <b>trace</b>   <b>flow</b> }	Sets the debug logging type.

## Managing Statistics Thresholds Using the GUI

- Step 1** On the menu bar, choose **Fabric > Fabric Policies**.
- Step 2** In the **Navigation** pane, click + to expand **Monitoring Policies**.
- Step 3** In the **Navigation** pane, expand the monitoring policy name (such as Default).
- Step 4** Click **Stats Collection Policies**.

- Step 5** In the **Stats Collection Policies** window, choose a **Monitoring Object** and **Stats Type** for which to set a threshold value..
- Step 6** In the **Work** pane, Click the + icon below **CONFIG THRESHOLDS**.
- Step 7** In the **THRESHOLDS FOR COLLECTION** window, click + to add a threshold.
- Step 8** In the **Choose a Property** window, choose a statistics type.
- Step 9** In the **EDIT STATS THRESHOLD** window, specify the following threshold values:
- Normal Value—A valid value of the counter.
  - Threshold Direction—Indicates whether the threshold is a maximum or minimum value.
  - Rising Thresholds (Critical, Major, Minor, Warning)—Triggered when the value exceeds the threshold.
  - Falling Thresholds (Critical, Major, Minor, Warning)—Triggered when the value drops below the threshold.
- Step 10** You can specify a set and reset value for rising and falling thresholds. The set value specifies when a fault is triggered; the reset value specifies when the fault is cleared.
- Step 11** Click **SUBMIT** to save the threshold value.
- Step 12** In the **THRESHOLDS FOR COLLECTION** window, click **CLOSE**.

## Statistics Troubleshooting Scenarios

The following table summarizes common statistics troubleshooting scenarios for the Cisco APIC.

Problem	Solution
The APIC does not enforce a configured monitoring policy	<p>The problem occurs when a monitoring policy is in place but the APIC does not perform a corresponding action, such as collecting the statistics or acting on a trigger threshold. Follow these steps to resolve the issue:</p> <ul style="list-style-type: none"> <li>• Verify that monPolDn points to the correct monitoring policy.</li> <li>• Ensure that the selectors are configured correctly and that there are no faults.</li> <li>• For Tenant objects, check the relation to the monitoring policy.</li> </ul>
Some configured statistics are missing.	<p>Follow these steps to resolve the issue:</p> <ul style="list-style-type: none"> <li>• Review the statistics that are disabled by default within the monitoring policy and collection policy.</li> <li>• Review the collection policy to determine if the statistics are disabled by default or disabled for certain intervals.</li> <li>• Review the statistics policy to determine if the statistics are disabled by default or disabled for certain intervals.</li> </ul> <p><b>Note</b> Except for fabric health statistics, 5 minute statistics are stored on the switch and are lost when the switch reboots.</p>

Problem	Solution
Statistics or history are not maintained for the configured time period.	<p>Follow these steps to resolve the issue:</p> <ul style="list-style-type: none"> <li>• Review the collection settings; if configured at the top level of the monitoring policy, the statistics can be overridden for a specific object or statistics type.</li> <li>• Review the collection policy assigned to the monitoring object. Confirm that the policy is present and review the administrative state, and history retention values.</li> <li>• Verify that the statistics type is configured correctly.</li> </ul>
Some statistics are not maintained for the full configured interval.	<p>Review whether the configuration exceeds the maximum historical record size. The limitations are as follows:</p> <ul style="list-style-type: none"> <li>• Switch statistics for 5 minute granularity are limited to 12 samples (1 hour of 5 minute granular statistics).</li> <li>• There is a hard limit of 1000 samples. For example, hourly granular statistics can be maintained for up to 41 days.</li> </ul>
An export policy is configured but the APIC does not export statistics.	<p>Follow these steps to resolve the issue:</p> <ul style="list-style-type: none"> <li>• Check the status object for the destination policy.</li> <li>• On the node that is expected to export the statistics check the export status object and look at the export status and details properties. Aggregated EPG stats are exported every 15 minutes from APIC nodes. Other statistics are exported from source nodes every 5 minutes. For example, if an EPG is deployed to two leaf switches and configured to export EPG aggregation parts, then those parts are exported from the nodes every 5 minutes.</li> <li>• Review whether the configuration exceeds the maximum number of export policies. The maximum number of statistics export policies is approximately equal to the number of tenants.</li> </ul> <p><b>Note</b> Each tenant can have multiple statistics export policies and multiple tenants can share the same export policy, but the total number number of policies is limited to approximately the number of tenants.</p>
5 Minute Statistics Fluctuate	The APIC system reports statistics every 5 minutes, sampled approximately every 10 seconds. The number of samples taken in 5 minutes may vary, because there are slight time variances when the data is collected. As a result, the statistics might represent a slightly longer or shorter time period. This is expected behavior.
Some historical statistics are missing.	For more information, see <a href="#">Statistics Cleanup</a> .

## Statistics Cleanup

The APIC and switches clean up statistics as follows:

- Switch—The switch cleans up statistics as follows:
  - 5 minute statistics on switches are purged if no counter value is reported for 5 minutes. This situation can occur when an object is deleted or statistics are disabled by a policy.
  - Statistics of larger granularity are purged if statistics are missing for more than one hour, which can occur when:
    - Statistics are disabled by a policy.
    - A switch is disconnected from an APIC for more than one hour.
  - The switch cleans up statistics for deleted objects after 5 minutes. If an object is recreated within this time, statistics counts remain unchanged.
  - Disabled object statistics are deleted after 5 minutes.
  - If the system state changes so that statistics reporting is disabled for 5 minutes, this switch cleans up statistics.
- APIC—The APIC cleans up objects including interfaces, EPGs, temperature sensors, and health statistics after one hour.

## Specifying Syslog Sources and Destinations

This section explains how to create syslog destination groups, a syslog source, and how to enable syslog to display in NX-OS CLI format using the REST API.

### About Syslog

During operation, a fault or event in the Cisco Application Centric Infrastructure (ACI) system can trigger the sending of a system log (syslog) message to the console, to a local file, and to a logging server on another system. A system log message typically contains a subset of information about the fault or event. A system log message can also contain audit log and session log entries.



---

**Note** For a list of syslog messages that the APIC and the fabric nodes can generate, see [http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci\\_syslog/ACI\\_SysMsg.html](http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci_syslog/ACI_SysMsg.html).

---

Many system log messages are specific to the action that a user is performing or the object that a user is configuring or administering. These messages can be the following:

- Informational messages, providing assistance and tips about the action being performed
- Warning messages, providing information about system errors related to an object, such as a user account or service profile, that the user is configuring or administering

In order to receive and monitor system log messages, you must specify a syslog destination, which can be the console, a local file, or one or more remote hosts running a syslog server. In addition, you can specify the minimum severity level of messages to be displayed on the console or captured by the file or host. The local file for receiving syslog messages is `/var/log/external/messages`.

A syslog source can be any object for which an object monitoring policy can be applied. You can specify the minimum severity level of messages to be sent, the items to be included in the syslog messages, and the syslog destination.

You can change the display format for the Syslogs to NX-OS style format.

Additional details about the faults or events that generate these system messages are described in the *Cisco APIC Faults, Events, and System Messages Management Guide*, and system log messages are listed in the *Cisco ACI System Messages Reference Guide*.



**Note** Not all system log messages indicate problems with your system. Some messages are purely informational, while others may help diagnose problems with communications lines, internal hardware, or the system software.

## Creating a Syslog Destination and Destination Group

This procedure configures syslog data destinations for logging and evaluation. You can export syslog data to the console, to a local file, or to one or more syslog servers in a destination group.

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

**Step 2** In the submenu bar, click **External Data Collectors**.

**Step 3** In the **Navigation** pane, expand **Monitoring Destinations**.

**Step 4** Right-click **Syslog** and choose **Create Syslog Monitoring Destination Group**.

**Step 5** In the **Create Syslog Monitoring Destination Group** dialog box, perform the following actions:

- a) In the group and profile **Name** field, enter a name for the monitoring destination group and profile.
- b) In the group and profile **Format** field, choose the format for Syslog messages.

The default is **aci**, or the RFC 5424 compliant message format, but you can choose to set it to the NX-OS style format instead.

- c) In the group and profile **Admin State** drop-down list, choose **enabled**.
- d) To enable sending of syslog messages to a local file, choose **enabled** from the Local File Destination **Admin State** drop-down list and choose a minimum severity from the Local File Destination **Severity** drop-down list.

The local file for receiving syslog messages is `/var/log/external/messages`.

- e) To enable sending of syslog messages to the console, choose **enabled** from the Console Destination **Admin State** drop-down list and choose a minimum severity from the Console Destination **Severity** drop-down list.
- f) Click **Next**.
- g) In the **Create Remote Destinations** area, click + to add a remote destination.

**Caution** Risk of hostname resolution failure for remote Syslog destinations, if the DNS server used is configured to be reachable over in-band connectivity. To avoid the issue, configure the Syslog server using the IP address, or if you use a hostname, ensure that the DNS server is reachable over an out-of-band interface.

**Step 6** In the **Create Syslog Remote Destination** dialog box, perform the following actions:

- a) In the **Host** field, enter an IP address or a fully qualified domain name for the destination host.
- b) (Optional) In the **Name** field, enter a name for the destination host.
- c) In the **Admin State** field, click the **enabled** radio button.



- d) (Optional) Choose a minimum severity **Severity**, a **Port** number, and a syslog **Facility**.

The **Facility** is a number that you can optionally use to indicate which process generated the message, and can then be used to determine how the message will be handled at the receiving end.

- e) From the **Management EPG** drop-down list, choose the management endpoint group.  
f) Click **OK**.

**Step 7** (Optional) To add more remote destinations to the remote destination group, click + again and repeat the steps in the **Create Syslog Remote Destination** dialog box

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

---

## Creating a Syslog Source

A syslog source can be any object for which an object monitoring policy can be applied.

### Before you begin

Create a syslog monitoring destination group.

---

**Step 1** From the menu bar and the navigation frame, navigate to a **Monitoring Policies** menu for the area of interest.

You can configure monitoring policies for tenants, fabric, and access.

**Step 2** Expand **Monitoring Policies**, then select and expand a monitoring policy.

Under **Fabric > Fabric Policies > Monitoring Policies > Common Policy** is a basic monitoring policy that applies to all faults and events and is automatically deployed to all nodes and controllers in the fabric. Alternatively, you can specify an existing policy with a more limited scope.

**Step 3** Under the monitoring policy, click **Callhome/SNMP/Syslog**.

**Step 4** In the **Work** pane, choose **Syslog** from the **Source Type** drop-down list.

**Step 5** From the **Monitoring Object** list, choose a managed object to be monitored.

If the desired object does not appear in the list, follow these steps:

- Click the Edit icon to the right of the **Monitoring Object** drop-down list.
- From the **Select Monitoring Package** drop-down list, choose an object class package.
- Select the checkbox for each object that you want to monitor.
- Click **Submit**.

**Step 6** In a tenant monitoring policy, if you select a specific object instead of **All**, a **Scope** selection appears.

In the **Scope** field, select a radio button to specify the system log messages to send for this object:

- **all**—Send all events and faults related to this object
- **specific event**—Send only the specified event related to this object. From the **Event** drop-down list, choose the event policy.
- **specific fault**—Send only the specified fault related to this object. From the **Fault** drop-down list, choose the fault policy.

**Step 7** Click + to create a syslog source.

**Step 8** In the **Create Syslog Source** dialog box, perform the following actions:

- a) In the **Name** field, enter a name for the syslog source.
- b) From the **Min Severity** drop-down list, choose the minimum severity of system log messages to be sent.
- c) In the **Include** field, check the checkboxes for the type of messages to be sent.
- d) From the **Dest Group** drop-down list, choose the syslog destination group to which the system log messages will be sent.
- e) Click **Submit**.

**Step 9** (Optional) To add more syslog sources, click + again and repeat the steps in the **Create Syslog Source** dialog box

## Enabling Syslog to Display in NX-OS CLI Format, Using the REST API

By default the Syslog format is RFC 5424 compliant. You can change the default display of Syslogs to NX-OS type format, similar to the following example:

```
apic1# moquery -c "syslogRemoteDest"

Total Objects shown: 1

syslog.RemoteDest
host : 172.23.49.77
adminState : enabled
childAction :
descr :
dn : uni/fabric/slgroup-syslog-mpod/rdst-172.23.49.77
epgDn :
format : nxos
forwardingFacility : local7
ip :
lcOwn : local
modTs : 2016-05-17T16:51:57.231-07:00
monPolDn : uni/fabric/monfab-default
name : syslog-dest
operState : unknown
port : 514
rn : rdst-172.23.49.77
severity : information
status :
uid : 15374
vrfId : 0
vrfName :
```

To enable the Syslogs to display in NX-OS type format, perform the following steps, using the REST API.

**Step 1** Enable the Syslogs to display in NX-OS type format, as in the following example:

```
POST https://192.168.20.123/api/node/mo/uni/fabric.xml
<syslogGroup name="DestGrp77" format="nxos">
<syslogRemoteDest name="slRmtDest77" host="172.31.138.20" severity="debugging"/>
</syslogGroup>
```

The **syslogGroup** is the Syslog monitoring destination group, the **sysLogRemoteDest** is the name you previously configured for your Syslog server, and the **host** is the IP address for the previously configured Syslog server.

**Step 2** Set the Syslog format back to the default RFC 5424 format, as in the following example:

```
POST https://192.168.20.123/api/node/mo/uni/fabric.xml
<syslogGroup name="DestGrp77" format="aci">
<syslogRemoteDest name="slRmtDest77" host="172.31.138.20" severity="debugging"/>
</syslogGroup>
```

---

## Discovering Paths and Testing Connectivity with Traceroute

This section lists the traceroute guidelines and restriction and explains how to perform a traceroute between endpoints.

### About Traceroute

The traceroute tool is used to discover the routes that packets actually take when traveling to their destination. Traceroute identifies the path taken on a hop-by-hop basis and includes a time stamp at each hop in both directions. You can use traceroute to test the connectivity of ports along the path between the generating device and the device closest to the destination. If the destination cannot be reached, the path discovery traces the path up to the point of failure.

A traceroute that is initiated from the tenant endpoints shows the default gateway as an intermediate hop that appears at the ingress leaf switch.

Traceroute supports a variety of modes, including:

- Endpoint-to-endpoint, and leaf-to-leaf (tunnel endpoint, or TEP to TEP)
- Endpoint-to-external-IP
- External-IP-to-endpoint
- External-IP-to-external-IP

Traceroute discovers all paths across the fabric, discovers point of exits for external endpoints, and helps to detect if any path is blocked.

### About Windows and Linux Traceroute

The **traceroute** command enables you to determine the path a packet takes in order to get to a destination from a given source by returning the sequence of hops the packet has traversed. This utility comes with your host operating system (for example, Linux or Microsoft (MS) Windows).

If you execute the **traceroute ip-address** command on a source device (such as a host, or a router acting as a host), it sends IP packets toward the destination with Time To Live (TTL) values that increment up to the maximum specified hop count. This is 30 by default. Typically, each router in the path towards the destination

decrements the TTL field by one unit while it forwards these packets. When a router in the middle of the path finds a packet with TTL = 1, it responds with an Internet Control Message Protocol (ICMP) "time exceeded" message to the source. This message lets the source know that the packet traverses that particular router as a hop




---

**Note** There are some differences with the way the **traceroute** command is implemented in the various operating systems as described in the following Linux and Windows sections.

---

### Linux

The TTL for the initial User Datagram Protocol (UDP) datagram probe is set to 1 (or the minimum TTL, as specified by user in the extended **traceroute** command). The destination UDP port of the initial datagram probe is set to 33434 (or as specified in the extended **traceroute** command output). The extended **traceroute** command is a variation of the ordinary **traceroute** command which allows the default values of the parameters used by the **traceroute** operation such as TTL and destination port number to be modified. The source UDP port of the initial datagram probe is randomized and has logical operator OR with 0x8000 (ensures a minimum source port of 0x8000). These steps illustrate what happens when the UDP datagram is launched:




---

**Note** The parameters are configurable. This example starts with n = 1 and finishes with n = 3.

---

1. The UDP datagram is dispatched with TTL = 1, destination UDP port= 33434, and the source port randomized.
2. The UDP destination port is incremented, the source UDP port is randomized, and the second datagram dispatched.
3. Step 2 is repeated for up to three probes (or as many times as requested in an extended **traceroute** command output). For each of the probes sent, you receive a "TTL exceeded" message, which is used to build a step-by-step path to the destination host.
4. TTL is incremented, and this cycle repeats with incremental destination port numbers, if the ICMP "time exceeded" message is received. You can also get one of these messages:
  - An ICMP type 3, code 3 ("destination unreachable," "port unreachable") message, which indicates that a host has been reached.
  - A "host unreachable," "net unreachable," "maximum TTL exceeded," or a "timeout" type of message, which means that the probe is resent.

Cisco routers send UDP probe packets with a random source port and an incremental destination port (to distinguish the different probes). Cisco routers send the ICMP message "time exceeded" back to the source from where the UDP/ICMP packet was received.

The Linux **traceroute** command is similar to the Cisco router implementation. However, it uses a fixed source port. The **-n** option in the **traceroute** command is used to avoid a request to a name server.



**Note** The CIMC controller on a UCS server does not respond to UDP-based traceroute messages. It responds only to ICMP-based traceroute. Windows traceroute, by default, sends ICMP-based messages. Linux (and Mac) traceroute, by default, sends UDP-based messages. The Linux (and Mac) traceroute will send ICMP-based messages if you use the -I (capital i) option.

Because of the Linux traceroute default, if you are troubleshooting an ACI network and need to send traceroute to the Cisco APIC, you must use Windows traceroute or specify ICMP-based traceroute.

### Windows

The MS Windows **tracert** command uses ICMP echo request datagrams instead of UDP datagrams as probes. ICMP echo requests are launched with incrementing TTL, and the same operation as described above occurs. The significance of using ICMP echo request datagrams is that the final hop does not rely on the response of an ICMP "unreachable" message from the destination host. It relies instead on an ICMP echo reply message.

The command syntax is:

```
tracert [-d] [-h maximum_hops] [-j computer-list] [-w timeout] target_name
```

This table explains the command parameters:

**Table 2:**

Parameter	Description
-d	Specifies not to resolve addresses to computer names.
-h maximum_hops	Specifies the maximum number of hops to search for a target.
-j computer-list	Specifies a loose source route along computer-list.
-w timeout	Waits the number of milliseconds specified by the timeout for each reply.
target_name	Name of the target computer.

## Traceroute Guidelines and Restrictions

- When the traceroute source or destination is an endpoint, the endpoint must be dynamic and not static. Unlike a dynamic endpoint (fv:CEp), a static endpoint (fv:StCEp) does not have a child object (fv:RsCEpToPathEp) that is required for traceroute.
- Traceroute works for IPv6 source and destinations but configuring source and destination IP addresses across IPv4 and IPv6 addresses is not allowed.
- See the *Verified Scalability Guide for Cisco ACI* document for traceroute-related limits.
- When an endpoint moves from one ToR switch to a different ToR switch that has a new MAC address (one that is different than the MAC address that you specified while configuring the traceroute policy), the traceroute policy shows "missing-target" for the endpoint. In this scenario you must configure a new traceroute policy with the new MAC address.

- When performing a traceroute for a flow involving the policy-based redirect feature, the IP address used by the leaf switch to source the time-to-live (TTL) expired message when the packet goes from the service device to the leaf switch may not always be the IP address of the bridge domain's switch virtual interface (SVI) of the service device. This behavior is cosmetic and does not indicate that the traffic is not taking the expected path.

## Performing a Traceroute Between Endpoints

---

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

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

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

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

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

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

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

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

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

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

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

## Using the Troubleshooting Wizard


The Troubleshooting Wizard allows you to understand and visualize how your network is behaving, which can ease your networking concerns should issues arise. For example, you might have two endpoints that are having intermittent packet loss, but you do not understand why. Using the Troubleshooting Wizard, you can evaluate the issue so that you can effectively resolve the issue rather than logging onto each machine that you suspect to be causing this faulty behavior.



This wizard allows you (the administrative user) to troubleshoot issues that occur during specific time frames for the chosen source and destination. You can define a time window in which you want to perform the debug, and you can generate a troubleshooting report that you can send to TAC.

## Getting Started with the Troubleshooting Wizard

Before you start using the Troubleshooting Wizard, you must be logged on as an Administrative user. Then, you must designate a source and destination and choose a time window for your troubleshooting session. The time window is used for retrieving events, fault records, deployment records, audit logs, and statistics.

As you navigate through the screens of the Troubleshooting Wizard, you have the option to take a screen shot

at any time and send it to a printer or save it as a PDF by clicking the **Print** icon (  ) at the top, right side

of the screen. There are also Zoom In and Zoom Out icons (   ) that you can use to modify your view of any screen.



### Note

- You cannot modify the source and destination after you click either **Generate Report** or **Submit**. If you want to change the source and destination information after you have entered it, you must delete the current session and start a new session.
- You cannot modify the description and time window on the first page of the wizard after you click **Submit**.
- You cannot use static IP address endpoints with the Troubleshooting Wizard.
- Any endpoints that you specify must be under an EPG.
- If you specify either the source or destination to be an external IP address, add the external subnets in the L3Out External EPG configuration. For information about configuring external subnets for external EPGs, see the "Routed Connectivity to External Networks" chapter of the *Cisco APIC Layer 3 Networking Configuration Guide*.

To set up your troubleshooting session information:

**Step 1** Choose **Operations > Visibility & Troubleshooting**.

The **Visibility & Troubleshooting** screen appears.

**Step 2** In the **Session Name** field, choose an existing troubleshooting session using the drop-down list or create a new session by entering a name.

**Step 3** In the **Session Type** drop-down list, choose the desired session type.

- **Endpoint to Endpoint:** The source and destination are both internal endpoints.

You should choose source and destination endpoints from the same tenant, or some of the troubleshooting functionality may be impacted, as explained later in this document. With this session type, you cannot use atomic counters when both endpoints connect to the same set of leaf switches.

- **Endpoint to External IP:** The source is an internal endpoint, while the destination is an external IP address.

- **External IP to Endpoint:** The source is an external IP address, while the destination is an internal endpoint.
- **External IP to External IP:** The source and destination are both external IP addresses. You can choose this type starting in the 3.2(6) release. With this session type, you cannot use traceroute, atomic counters, nor latency.

**Step 4** (Optional) Enter a description in the **Description** field to provide additional information.

**Step 5** Enter the source information in the **Source** area.

- If you chose a session type of **Endpoint to Endpoint** or **Endpoint to External IP**, enter a MAC, IPv4, or IPv6 address, or a VM name, then click **Search**.

You can enter a MAC address only if the session type is **Endpoint to Endpoint** and both endpoints' MAC address do not have any IP addresses learned from them.

A box appears that displays one or more rows with detailed information to help you make a selection. Each row shows that the IP address (in the **IP** column) you entered is in a specific endpoint group (in the **EPG** column), which belongs to a certain application (in the **Application** column), which is in a particular tenant (in the **Tenant** column). The leaf switch number, FEX number, and port details are shown in the **Learned At** column.

- If you chose a session type of **External IP to Endpoint**, enter the external IP address.
- If you chose a session type of **External IP to External IP**, enter the external IP address and distinguished name of the external Layer 3 outside network.

**Step 6** Enter the destination information in the **Destination** area.

- If you chose a session type of **Endpoint to Endpoint** or **External IP to Endpoint**, enter a MAC, IPv4, or IPv6 address, or a VM name, then click **Search**.

You can enter a MAC address only if the session type is **Endpoint to Endpoint** and both endpoints' MAC address do not have any IP addresses learned from them.

A box appears that displays one or more rows with detailed information to help you make a selection. Each row shows that the IP address (in the **IP** column) you entered is in a specific endpoint group (in the **EPG** column), which belongs to a certain application (in the **Application** column), which is in a particular tenant (in the **Tenant** column). The leaf switch number, FEX number, and port details are shown in the **Learned At** column.

- If you chose a session type of **Endpoint to External IP**, enter the external IP address.
- If you chose a session type of **External IP to External IP**, enter the external IP address and distinguished name of the external Layer 3 outside network.

**Step 7** In the **Time Window** area, specify a time window.

The **Time Window** is used for debugging an issue that occurred during a specific time frame in the past, and is used for retrieving events, all records, deployment records, audit logs, and statistics. There are two sets of windows: one for all records and one for individual leaf switches (or nodes).

By default, you can specify a rolling time window based on any number of minutes that you specify in the **Latest Minutes** field. The default is 240 minutes. The session contains data for the past number of minutes that you specify that precede the time that you created the session.

If you put a check in the **Use fixed time** box, you can specify a fixed time window for the session in the **From** and **To** fields. The session contains data starting from the **From** time to the **To** time.

**Step 8** Click **Submit** to begin your troubleshooting session.



After a short delay, the topology diagram for your troubleshooting session appears.

---

## Generating Troubleshooting Reports

You can generate a troubleshooting report in several formats, including JSON, XML, PDF, and HTML. Once you select a format, you can download the report (or schedule a download of the report) and use it for offline analysis or you can send it to TAC so that a support case can be created.

To generate a troubleshooting report:

---

- Step 1** From the bottom right corner of the screen, click **GENERATE REPORT**. The **Generate Report** dialog box appears.
  - Step 2** Choose an output format from the Report Format drop-down menu (**XML**, **HTML**, **JSON**, or **PDF**).
  - Step 3** If you want to schedule the download of the report to happen immediately, click the **Now > SUBMIT**. An **Information** box appears indicating where to obtain the report once it has been generated.
  - Step 4** To schedule the generation of the report for a later time, choose a schedule by clicking **Use a scheduler > Scheduler** drop-down menu then choose either an existing schedule or create a new one by clicking **Create Scheduler**. The **CREATE TRIGGER SCHEDULE** dialog appears.
  - Step 5** Enter information for the **Name**, **Description** (optional), and **Schedule Windows** fields.  
**Note** For more information on how to use the **SCHEDULER**, please refer to the online help.
  - Step 6** Click **SUBMIT**.
- 

The reports take some time to generate (from a couple of minutes to up to ten minutes), depending on the size of the fabric and how many faults or events exist. A status message displays while the report is being generated. To retrieve and view the troubleshooting report, click **SHOW GENERATED REPORTS**.

Supply the credentials (**User Name** and **Password**) of the server in the **Authentication Required** window. The troubleshooting report is then downloaded locally to your system.

The **ALL REPORTS** window appears showing a list of all the reports that have been generated, including the one you just triggered. From there, you can click the link to either download or immediately view the report, depending on the output file format you chose (for example, if the file is a PDF, it may open immediately in your browser).


## Topology in the Troubleshooting Wizard

This section explains the topology in the Troubleshooting Wizard. The topology shows how the source and destination are connected to the fabric, what the network path is from the source to the destination, and what the intermediate switches are.

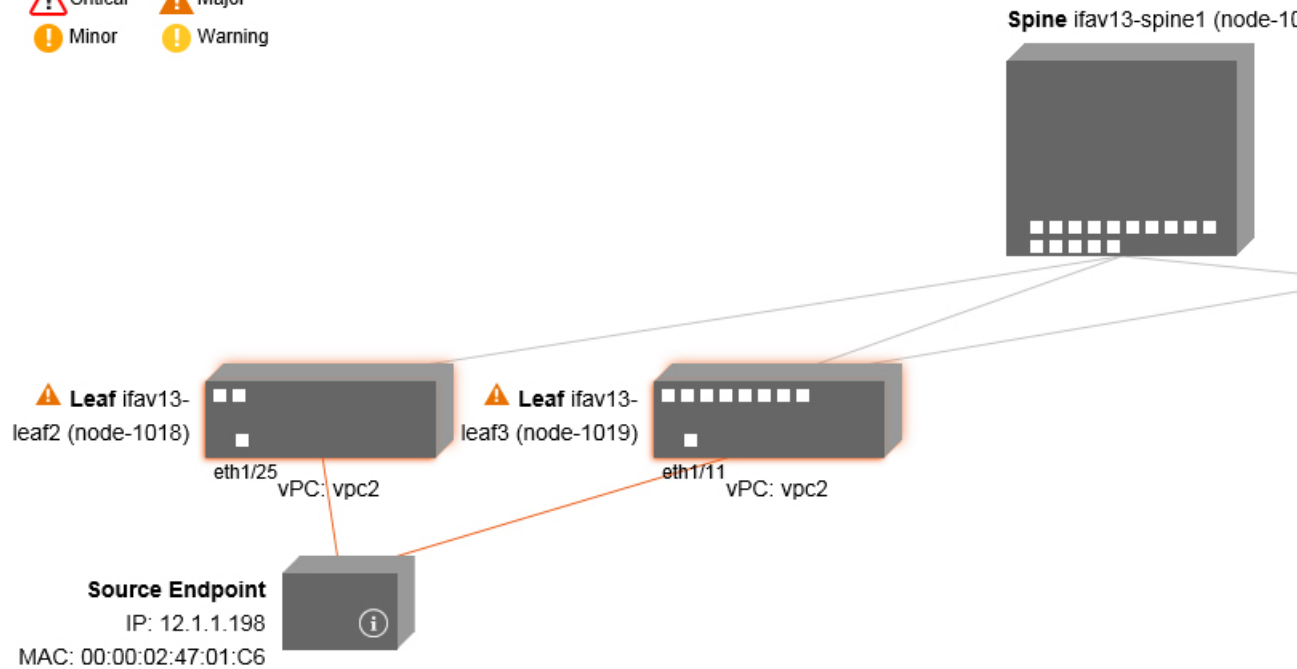
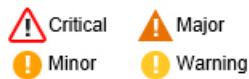
The source is displayed on the left side of the topology and the destination is on the right, as shown in the following wizard topology diagram.



**Note** This wizard topology only shows the leaf switches, spine switches, and FEXes of the devices involved in the traffic from the source to the destination. However, there may be many other leaf switches (tens or hundreds of leaf switches and many other spine switches) that exist.

This topology also shows links, ports, and devices. If you hover over the  icon), you can see the tenant that the source or destination belongs to, which application it belongs to, and the traffic encapsulation it is using (such as VLAN).

There is a color legend on the left side of the screen (shown as follows) that describes the severity levels associated with each color in the topology diagram (for example, critical versus minor).



Hovering over items such as boxes or ports in the topology provides more detailed information. If the port or link has a color, this means that there is a problem for you to troubleshoot. For example, if the color is red or orange, this indicates that there is a fault on a port or link. If the color is white, then there are no faults that exist. If the link has a number in a circle, it indicates how many parallel links between the same two nodes are affected by a fault of the severity given by the color of the circle. Hovering over a port allows you to see which port is connected to the source.

Right-clicking on a leaf switch allows you to access the console of the switch. A pop-up window appears that allows you to log into that device.

**Note**





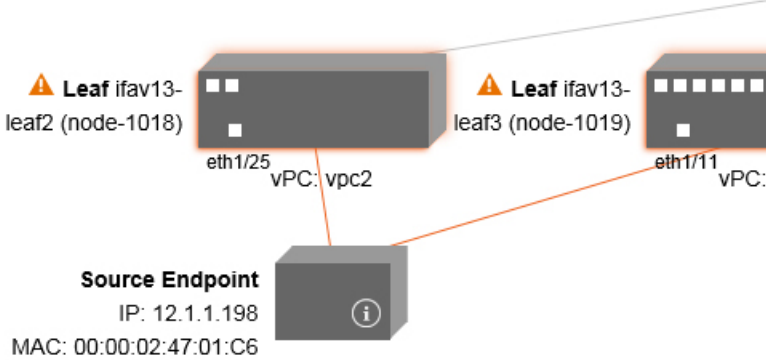


- If there are Layer 4 to Layer 7 services (firewall and load balancer), they will be shown in the topology as well.
- For a topology with the load balancer, the destination is expected to be the virtual IP (VIP) address.
- When the source or destination is behind an ESX server, the ESX is shown in the topology.

## Using the Faults Troubleshooting Screen

This procedure describes how to use the faults of the Troubleshooting Wizard.

### Procedure

	Command or Action	Purpose
<b>Step 1</b>	Click <b>Faults</b> in the <b>Navigation</b> pane to begin using the <b>Faults</b> troubleshooting screen.	<p>The <b>Faults</b> screen shows the topology that connects the source and destination that you previously selected as well as the faults that were found. Only faults for the designated communication are shown. Wherever there are faults, they are highlighted in a certain color to convey the severity. Refer to the color legend at the top of the screen to understand the severity levels associated with each color. White boxes indicate that there are no issues to troubleshoot in that particular area.</p> <p>This topology also shows the relevant leaf switches, spine switches, and FEXes for your troubleshooting session. Hovering over items, such as leaf switches, spine switches, and FEXes, or clicking on faults provides more detailed information for analysis.</p>

	Command or Action	Purpose
		<p>  Critical     Major   Minor     Warning </p>  <p>  Leaf ifav13-leaf2 (node-1018)     Leaf ifav13-leaf3 (node-1019) </p> <p> Source Endpoint  IP: 12.1.1.198  MAC: 00:00:02:47:01:C6 </p>
<b>Step 2</b>	Click on a fault to display a dialog box with the <b>Drop Stats</b> , <b>Contract Drops</b> , and <b>Traffic Stats</b> tabs that contain more detailed information for analysis.	

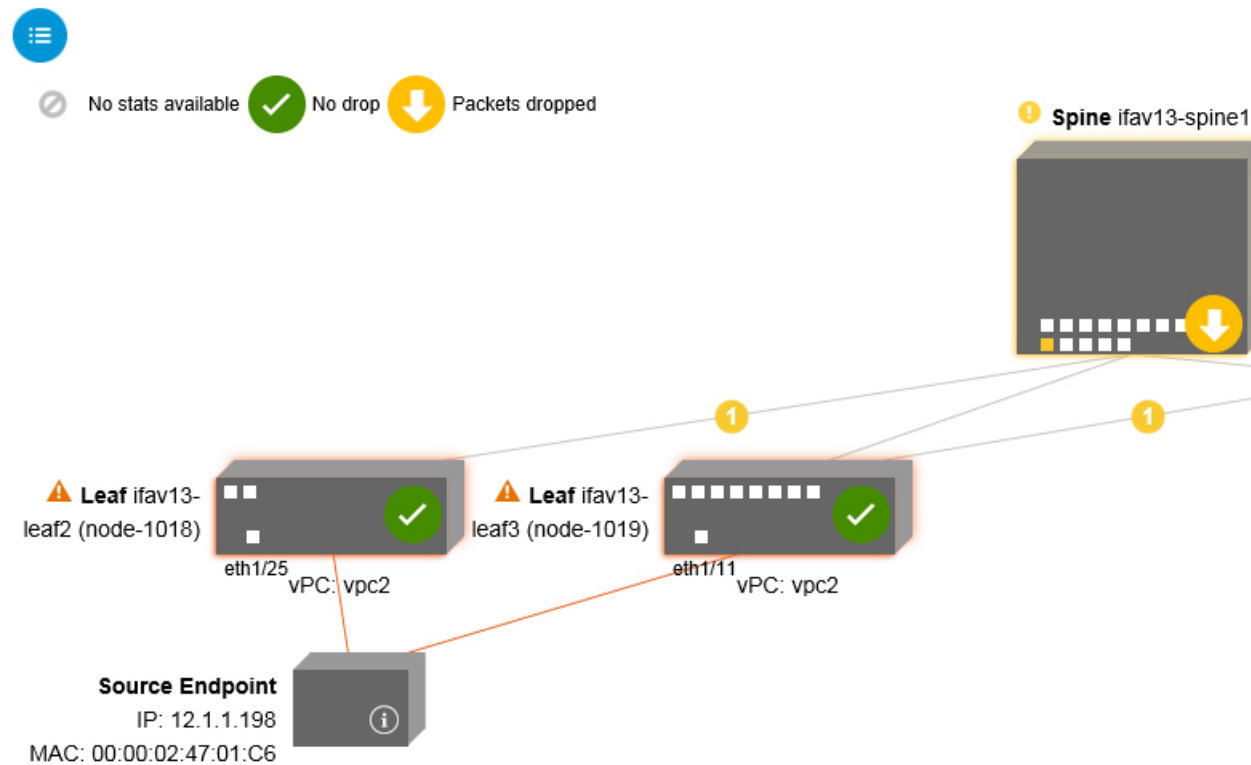
**Related Topics**

[Using the Drop/Statistics Troubleshooting Screen](#), on page 94

## Using the Drop/Statistics Troubleshooting Screen

Click **Drop/Stats** in the **Navigation** pane to begin using the **Drop/Stats** troubleshooting screen.

The **Drop/Stats** window displays the topology with all the statistics from the drops so that you can clearly see where drops exist or not. You can click on any drop image to see more information for analysis.



Once you click a drop image, there are three tabs at the top of the **Drop/Stats** screen, and the statistics shown are localized to that particular leaf or switch.

The three statistics tabs are:

- **DROP STATS**

This tab shows the statistics for drop counters. The packets that are dropped at various levels are shown here.




---

**Note** By default, counters with zero values are hidden but the user can choose to see all the values.

---

- **CONTRACT DROPS**

This tab shows a list of the contract drops that have occurred, which are individual packet logs (ACL logs), and shows information for each packet such as the **Source Interface**, **Source IP address**, **Source Port**, **Destination IP address**, **Destination Port**, and **Protocol**.




---

**Note** Not every packet is displayed here.

---

- **TRAFFIC STATS**


This tab shows the statistics that indicate ongoing traffic. These are how many packets have been transferring.




---

**Note** By default, counters with zero values are hidden but the user can choose to see all the values.

---

You can also view all of the statistics for all managed objects at once by clicking the All icon () located in the top, left corner of the screen.

You also have the option to pick zero or non-zero drops. Checking the box for **Show stats with zero values** (located in the top, left corner of the screen) allows you to see all the existing drops. The fields for **Time**, **Affected Object**, **Stats**, and **Value** become populated with data for all the zero values.

If you do not check the **Show stats with zero values** box, you will see results with non-zero drops.




---

**Note** The same logic applies if you click the **All** icon. All three tabs (**DROP STATS**, **CONTRACT DROPS**, and **TRAFFIC STATS**) are also available and have the same type of information appearing.

---

#### Related Topics

[Using the Contracts Troubleshooting Screen](#), on page 96

## Using the Contracts Troubleshooting Screen

Click **Contracts** in the **Navigation** pane to begin using the **Contracts** troubleshooting screen.

The **Contracts** troubleshooting screen displays the contracts that are applicable from the Source to the Destination and from the Destination to the Source.

Each one of the blue table heading rows indicates a filter. There are multiple rows under each filter that indicate multiple filter entries (**Protocol**, **L4 Src**, **L4 Dest**, **TCP Flags**, **Action**, **Nodes**, and **Hits**) for a particular leaf or switch.

Hovering over the certificate icon, shows you the contract name and the contract filter name. The text appearing on the right side of each blue table heading row (or filter) tells what type of contract it is, for example:

- Epg to Epg
- BD Allow
- Any to Any
- Context Deny


These contracts are categorized from the Source to the Destination and from the Destination to the Source.




---

**Note** The hits shown for each filter are cumulative (that is, the total hits for that contract hit, contract filter, or rule are shown for each particular leaf.) Statistics are refreshed automatically every (one) minute.

---

You can get policy information by hovering over the Information () icon. You can also see which EPGs are being referred to.



---

**Note** If there are no contracts between the endpoints, this will be indicated with a **There is no contract data** pop-up.

---

#### Related Topics

[Using the Events Troubleshooting Screen](#), on page 97


## Using the Events Troubleshooting Screen

Click **Events and Audits** in the **Navigation** pane to begin using the **Events and Audits** troubleshooting screen.

If you click on an individual leaf or spine switch, you can see more detailed information about that individual event.

There are two tabs available: **EVENTS** and **DEPLOYMENT RECORDS**.

- **EVENTS** show event records for any changes that have occurred in systems (such as physical interfaces or VLANs, for example). There are individual events listed for each particular leaf. You can sort these events based on **Severity**, **Affected Object**, **Creation Time**, **Cause**, and **Description**.
- **DEPLOYMENT RECORDS** show the deployment of policies on physical interfaces, VLANs, VXLANs, and L3 CTXs. These records show the time when a VLAN was placed on a leaf because of the epg.

If you click the **All** icon () for the **All Changes** screen, you can see all the events indicating any changes that have occurred during your specified time interval (or troubleshooting session).

There are three tabs in the **All Changes** screen, including:

- **AUDITS**  
Audits do not have a leaf association, which is why they are only available in the **All Changes** screen.
- **EVENTS** (described above)
- **DEPLOYMENT RECORDS** (described above)

#### Related Topics

[Using the Traceroute Troubleshooting Screen](#), on page 97

## Using the Traceroute Troubleshooting Screen

Click **Traceroute** in the **Navigation** pane to begin using the **Traceroute** troubleshooting screen.

To create and run a traceroute for troubleshooting:

1. In the **TRACEROUTE** dialog box, choose a destination port from the **Destination Port** drop-down menu.
2. Choose a protocol from the **Protocol** pull-down menu. The options supported include:

- **icmp**—This protocol is uni-directional, in that it does a traceroute from the Source leaf to the Destination endpoint only.
- **tcp**—This protocol is also bi-directional, as described above for the **udp** protocol.
- **udp**—This protocol is bi-directional, in that it does a traceroute from the Source leaf to the Destination endpoint, then from the Destination leaf back to the Source endpoint.




---

**Note** UDP, TCP and ICMP are the only supported protocols for IPv4. For IPv6, only UDP is supported.

---

3. Once you create a traceroute, click the **Play** (or Start) button to start the traceroute.




---

**Note** When you press the **Play** button, the policies are created on the system and a **Warning** message appears.

---

4. Click **OK** to proceed and the traceroute starts to run.
5. Click the **Stop** button to end the traceroute.




---

**Note** When you press the **Stop** button, the policies are removed from the system.

---

Once the traceroute completes, you can see where it was launched and what the result was. There is a pull-down menu next to **Traceroute Results** that shows where the traceroute was launched (from the Source to the Destination or from the Destination to the Source).

The result is also shown in the **Traceroute** dialog, which includes information for **Running Time**, **Traceroute Status**, **Destination Port**, and **Protocol**.

The results are represented by green and/or red arrows. A green arrow is used to represent each node in the path that responded to the traceroute probes. The beginning of a red arrow represents where the path ends as that's the last node that responded to the traceroute probes. You don't choose which direction to launch the traceroute. Instead, the traceroute is always started for the session. If the session is:

- EP to external IP or external IP to EP, the traceroute is always launched from EP to external IP.
- EP to EP and protocol is ICMP, the traceroute is always launched from the source to the destination.
- EP to EP and protocol is UDP/TCP, the traceroute is always bidirectional.



- 
- Note**
- The **Traceroute Results** drop-down menu can be used to expose/visualize the results for each direction for scenario #3 above. In scenarios #1 and #2, it's always greyed out.
  - If the **Traceroute Status** shows as incomplete, this means you are still waiting for part of the data to come back. If the **Traceroute Status** shows as **complete**, then it is actually complete.
- 

#### Related Topics

[Using the Atomic Counter Troubleshooting Screen](#), on page 99



## Using the Atomic Counter Troubleshooting Screen

Click **Atomic Counter** in the **Navigation** pane to begin using the **Atomic Counter** troubleshooting screen.

The Atomic Counter screen is used to take source and destination information and create a counter policy based on that. You can create an atomic counter policy between two endpoints and monitor the traffic going back and forth from the Source to the Destination and from the Destination to the Source. You can determine how much traffic is going through and especially determine if any anomalies (drops or excess packets) are reported between the source and destination leaves.

There are **Play** (or **Start**) and **Stop** buttons at the top of the screen so that you can start and stop the atomic counter policy at any point and can count the packets that are being sent.



---

**Note** When you press the **Play** button, the policies are created on the system and the packet counter starts. When you press the **Stop** button, the policies are removed from the system.

---

The results are shown in two different formats. You can view them in either a brief format, which includes a summary, or in a longer format (by clicking on the **Expand** button). Both brief and expanded formats show both directions. The expanded format shows the cumulative counts plus the counts for each of the latest 30s intervals, while the brief format only shows the counts for cumulative and last interval.

### Related Topics

[Using the SPAN Troubleshooting Screen](#), on page 99

## Using the SPAN Troubleshooting Screen

Click **SPAN** in the **Navigation** pane to begin using the **SPAN** troubleshooting screen.

Using this screen, you can span (or mirror) bi-directional traffic and redirect it to the analyzer. In a SPAN session, you are making a copy and sending it to the analyzer.

This copy goes to a particular host (the analyzer IP address) and then you can use a software tool such as Wireshark to view the packets. The session information has source and destination information, session type, and the timestamp range.



---

**Note** When you press the **Play** button, the policies are created on the system. When you press the **Stop** button, the policies are removed from the system.

---



---

**Note** For a list of Troubleshooting Wizard CLI commands, see the *Cisco APIC Command-Line Interface User Guide*.

---

## Creating a SPAN Session Using the Cisco APIC Troubleshooting CLI

This section demonstrates how to use the Cisco APIC troubleshooting CLIs to create a SPAN session.

**Step 1** `troubleshoot node session <session_name> nodename <node_id>`

To create a node-level session (global drop):

**Example:**

```
apic1(config)# troubleshoot node session 301-GD-APIC nodeid 301
```

**Step 2** `troubleshoot node session <session_name> nodename <node_id> interface ethernet <interface>`

To create an interface-level session:

**Example:**

```
apic1(config)# troubleshoot node session 301-GD-APIC nodeid 301 interface eth1/3
```

**Step 3** `troubleshoot node session <session_name> monitor destination apic_ip srcipprefix <ip_prefix> drop enable erspan-id[optional]`

To specify the destination as Cisco APIC and enable SPAN on drop:

**Example:**

```
apic1(config)# troubleshoot node session 301-GD-APIC monitor destination apic srcipprefix 13.13.13.13 drop enable
```

**Step 4** `troubleshoot node session <session_name> monitor destination tenant tenant application <app> destip <dest_ip>srcipprefix<ip_prefix>drop enable erspan-id[optional]`

To specify an ERSPAN destination and enable SPAN on drop:

**Example:**

```
apic1(config)# troubleshoot node session 301-GD-APIC monitor destination tenant ERSPAN application A1 epg E1 destip 179.10.10.179 srcipprefix 31.31.13.31 drop enable
```

To check the SPAN-on-drop packets on the Cisco APIC when it is set as destination:

- a. Disable the SPAN-on-drop session:

```
apic1(config)# no troubleshoot node session 301-GD-APIC monitor
```

- b. Go to the drop-stats directory and check the DropPackets\_\*.pcap file:

```
/data2/techsupport/troubleshoot/node/Session_name/span_capture/drop-stats/DropPackets_*.pcap
```

## L4 - L7 Services Validated Scenarios

The Troubleshooting Wizard allows you to provide two endpoints and see the corresponding topology between those endpoints. When L4 - L7 services exist between the two endpoints in the topology, you are able to view these as well.

This section describes the L4 - L7 scenarios that have been validated for this release. Within the L4 - L7 services, the number of topologies is very high, which means that you can have different configurations for firewalls, load balancers, and combinations of each. If a firewall exists between the two endpoints in the topology, the Troubleshooting Wizard retrieves the firewall data and connectivity from the firewall to the leafs. If a load balancer exists between the two endpoints, you can retrieve and view information up to the load balancer but not up to the server.

The following table shows the L4 - L7 service scenarios that were validated for the Troubleshooting Wizard:

Scenario	1	2	3	4	5	6
Number of Nodes	1	1	2	1	1	2
Device	GoTo FW (vrf split)	GoTo SLB	GoTo,GoTo FW,SLB	FW-GoThrough	SLB-GoTo	FW, SLB (GoThrough, GoTo)
Number of Arms	2	2	2	2	2	2
Consumer	EPG	EPG	EPG	L3Out	L3Out	L3Out
Provider	EPG	EPG	EPG	EPG	EPG	EPG
Device Type	VM	VM	VM	physical	physical	physical
Contract Scope	tenant	context	context	context	context	global
Connector Mode	L2	L2	L2, L2	L3, L2	L3	L3 / L2,L3
Service Attach	BSW	BSW	DL/PC	regular port	vPC	regular port
Client Attach	FEX	FEX	FEX	Regular Port	Regular Port	regular port
Server Attach	vPC	vPC	vPC	regular port	regular port	regular port

## List of APIs for Endpoint to Endpoint Connections

The following is a list of the available Troubleshooting Wizard APIs for EP to EP (endpoint to endpoint) connections:

- [interactive API](#), on page 102
- [createsession API](#), on page 103
- [modifysession API](#), on page 104
- [atomiccounter API](#), on page 104
- [traceroute API](#), on page 104
- [span API](#), on page 105
- [generatereport API](#), on page 106
- [schedulingreport API](#), on page 106
- [getreportstatus API](#), on page 107
- [getreportslist API](#), on page 107
- [getsessionslist API](#), on page 107
- [getsessiondetail API](#), on page 108
- [deletesession API](#), on page 108
- [clearreports API](#), on page 109

- [contracts API, on page 109](#)

## interactive API

To create an endpoint (ep) to endpoint interactive troubleshooting session, use the **interactive** API. The module name is **troubleshoot.eptoeputils.topo** and the function is **getTopo**. The required argument (**req\_args**) for the interactive API is **- session**.

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax	Description	Optional Arguments (opt_args)	Description
		- srcep	Source endpoint name
		- dstep	Destination endpoint name
		- srcip	Source endpoint IP address
		- dstip	Destination endpoint IP address
		- srcmac	Source endpoint MAC
		- dstmac	Destination endpoint MAC
		- srcextip	L3 external source IP address
		- dstextip	L3 external destination IP address
		- starttime	Start time of the troubleshooting session
		- endtime	End time of the troubleshooting session
		- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
		- description	Description about the session
		- scheduler	Scheduler name for report generation
		- srcepid	Obsolete
		- dstepid	Obsolete
		- include	Obsolete
		- format	Format of report to be generated
		- ui	Used internally (ignore)
		- sessionurl	Location of the report
		-action	Start/stop/status etc. for traceroute/atomiccounter
		- mode	Used internally

-_dc	Used internally
-ctx	Used internally

## createsession API

To create an endpoint (ep) to endpoint troubleshooting session, use the **createsession** API. The module name is **troubleshoot.eptoeptutils.session** and the function is **createSession**.

The required argument (**req\_args**) for the createsession API is - **session** (session name).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax	Description	Optional Arguments (opt_args)	Description
		- srcep	Source endpoint name
		- dstep	Destination endpoint name
		- srcip	Source endpoint IP address
		- dstip	Destination endpoint IP address
		- srcmac	Source endpoint MAC
		- dstmac	Destination endpoint MAC
		- srcectip	L3 external source IP address
		- dstectip	L3 external destination IP address
		- starttime	Start time of the troubleshooting session
		- endtime	End time of the troubleshooting session
		- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
		- description	Description about the session
		- format	Format of report to be generated
		- ui	Used internally (ignore)
		-action	Start/stop/status etc. for traceroute/atomiccounter
		- scheduler	
		- srctenant	Name of the tenant for the source endpoint
		- srcapp	Name of the app for the source endpoint
		- srcepg	Name of the endpoint group for the source endpoint
		- dsttenant	Name of the tenant for the destination endpoint

- dstapp	Name of the app for the destination endpoint
- dstepg	Name of the endpoint group for the destination endpoint
- mode	Used internally

## modifysession API

To modify an endpoint (ep) to endpoint troubleshooting session, use the **modifysession** API. The module name is **troubleshoot.eptoeputils.topo** and the function is **modifySession**.

The required arguments (**req\_args**) for the modifysession API are **- session** (session name) and **- mode**.

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- starttime	Start time of the troubleshooting session
	- endtime	End time of the troubleshooting session
	- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
	- description	Description about the session

## atomiccounter API

To create an endpoint (ep) to endpoint atomic counter session, use the **atomiccounter** API. The module name is **troubleshoot.eptoeputils.atomiccounter** and the function is **manageAtomicCounterPols**.

The required arguments (**req\_args**) for the atomiccounter API include:

- - session
- - action
- - mode



**Note** There are no optional arguments (**opt\_args**) for the atomiccounter API.

## traceroute API

To create an endpoint (ep) to endpoint traceroute session using the API, use the **traceroute** API. The module name is **troubleshoot.eptoeputils.traceroute** and the function is **manageTraceroutePols**.

The required arguments (**req\_args**) for the traceroute API include:

- - session (session name)
- - action (start/stop/status)

- - mode

Syntax Description	Optional Arguments (opt_args)	Description
	- protocol	Protocol name
	- dstport	Destination port name

## span API

To create an endpoint (ep) to endpoint span troubleshooting session, use the **span** API. The module name is **troubleshoot.eptoeputils.span** and the function is **monitor**.

The required arguments (**req\_args**) for the span API include:

- - session (session name)
- - action (start/stop/status)

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name
	- dstep	Destination endpoint name
	- srcip	Source endpoint IP address
	- dstip	Destination endpoint IP address
	- srcmac	Source endpoint MAC
	- dstmac	Destination endpoint MAC
	- srcextip	L3 external source IP address
	- dstextip	L3 external destination IP address
	- starttime	Start time of the troubleshooting session
	- endtime	End time of the troubleshooting session
	- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
	- description	Description about the session
	- scheduler	Scheduler name for report generation
	- srcepid	Obsolete
	- dstepid	Obsolete
	- include	Obsolete

- format	Format of report to be generated
- ui	Used internally (ignore)
- sessionurl	Location of the report
-action	Start/stop/status etc. for traceroute/atomiccounter
- srctenant	Name of the tenant for the source endpoint
- srcapp	Name of the app for the source endpoint
- srcepg	Name of the endpoint group for the source endpoint
- dsttenant	Name of the tenant for the destination endpoint
- dstapp	Name of the app for the destination endpoint
- dstepg	Name of the endpoint group for the destination endpoint
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## generatereport API

To generate a troubleshooting report using the API, use the **generatereport** API. The module name is **troubleshoot.eptoeputils.report** and the function is **generateReport**.

The required arguments (**req\_args**) for the generatereport API are **- session** (session name) and **- mode**.

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- include	Obsolete
	- format	Format of report to be generated

## schedulereport API

To schedule the generation of a troubleshooting report using the API, use the **schedulereport** API. The module name is **troubleshoot.eptoeputils.report** and the function is **scheduleReport**. The required argument (**req\_args**) for the schedulereport API is **- session**

The required arguments (**req\_args**) for the schedulereport API include:

- - session (session name)
- - scheduler (scheduler name)
- - mode



The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- starttime	Start time of the troubleshooting session
	- endtime	End time of the troubleshooting session
	- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
	- include	Obsolete
	- format	Format of report to be generated
	- action	Start/stop/status etc. for traceroute/atomiccounter

## getreportstatus API

To get the status of a generated report using the API, use the **getreportstatus** API. The module name is **troubleshoot.eptoeputils.report** and the function is **getStatus**.

The required arguments (**req\_args**) for the getreportstatus API include:

- - session (session name)
- - sessionurl (session URL)
- - mode



**Note** There are no optional arguments (**opt\_args**) for the getreportstatus API.

## getreportslist API

To get a list of generated reports using the API, use the **getreportslist** API. The module name is **troubleshoot.eptoeputils.report** and the function is **getReportsList**.

The required arguments (**req\_args**) for the getreportslist API are **- session** (session name) and **- mode**.



**Note** There are no optional arguments (**opt\_args**) for the getreportslist API.

## getsessionslist API

To get a list of troubleshooting sessions using the API, use the **getsessionslist** API. The module name is **troubleshoot.eptoeputils.session** and the function is **getSessions**.

The required argument (**req\_args**) for the getsessionlist API is **- mode**.




---

**Note** There are no optional arguments (**opt\_args**) for the getsessionlist API.

---

## getsessiondetail API

To get specific details about a troubleshooting session using the API, use the **getsessiondetail** API. The module name is **troubleshoot.eptoeputils.session** and the function is **getSessionDetail**.

The required arguments (**req\_args**) for the getsessiondetail API are - **session** (session name) and - **mode**.




---

**Note** There are no optional arguments (**opt\_args**) for the getsessiondetail API.

---

## deletesession API

To delete a particular troubleshooting session using the API, use the **deletesession** API. The module name is **troubleshoot.eptoeputils.session** and the function is **deleteSession**.

The required argument (**req\_args**) for the deletesession API is - **session** (session name).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name
	- dstep	Destination endpoint name
	- srcip	Source endpoint IP address
	- dstip	Destination endpoint IP address
	- srcmac	Source endpoint MAC
	- dstmac	Destination endpoint MAC
	- srcextip	L3 external source IP address
	- dstextip	L3 external destination IP address
	- starttime	Start time of the troubleshooting session
	- endtime	End time of the troubleshooting session
	- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
	- description	Description about the session
	- scheduler	Scheduler name for report generation

- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	Format of report to be generated
- ui	Used internally (ignore)
- sessionurl	Location of report
- action	Start/stop/status etc. for traceroute/atomiccounter
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## clearreports API

To clear the list of generated reports using the API, use the **clearreports** API. The module name is **troubleshoot.eptoeputils.report** and the function is **clearReports**.

The required arguments (**req\_args**) for the clearreports API are- **session** (session name) and - **mode**.



**Note** There are no optional arguments (**opt\_args**) for the clearreports API.

## contracts API

To get contracts information using the API, use the **contracts** API. The module name is **troubleshoot.eptoeputils.contracts** and the function is **getContracts**.

The required arguments (**req\_args**) for the contracts API are- **session** (session name) and -**mode**.

There are no optional arguments (**opt\_args**) for the contracts API.

## List of APIs for Endpoint to Layer 3 External Connections

The following is a list of the available Troubleshooting Wizard APIs for EP to EP (endpoint to endpoint) connections:

- [interactive API, on page 110](#)
- [modifysession API, on page 111](#)
- [atomiccounter API, on page 112](#)
- [traceroute API, on page 113](#)
- [span API, on page 114](#)

- [generatereport API](#), on page 114
- [schedulesreport API](#), on page 116
- [getreportstatus API](#), on page 107
- [getreportslist API](#), on page 107
- [clearreports API](#), on page 109
- [createsession API](#), on page 110
- [getsessionslist API](#), on page 117
- [getsessiondetail API](#), on page 118
- [deletesession API](#), on page 119
- [contracts API](#), on page 120
- [ratelimit API](#), on page 121
- [l3ext API](#), on page 121

## interactive API

To create an endpoint (ep) to Layer 3 (L3) external interactive troubleshooting session, use the **interactive** API. The module name is **troubleshoot.epextutils.epext\_topo** and the function is **getTopo**. The required arguments (**req\_args**) for the interactive API are **- session**, **- include**, and **- mode**.

The following table shows the optional argument (**opt\_args**):

Syntax Description	Optional Arguments (opt_args)	Description
	- refresh	

## createsession API

To create an endpoint (Ep) to Layer 3 (L3) external troubleshooting session using the API, use the **createsession** API. The module name is **troubleshoot.epextutils.epextsession** and the function is **createSession**. The required argument (**req\_args**) for the createsession API is **- session** (session name).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name
	- dstep	Destination endpoint name
	- srcip	Source endpoint IP address
	- dstip	Destination endpoint IP address
	- srcmac	Source endpoint MAC

- dstmac	Destination endpoint MAC
- srcextip	L3 external source IP address
- dstextip	L3 external destination IP address
- starttime	Start time of the troubleshooting session
- endtime	End time of the troubleshooting session
- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
- description	Description about the session
- scheduler	Scheduler name for report generation
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	Format of report to be generated
- ui	Used internally (ignore)
- sessionurl	Location of the report
-action	Start/stop/status etc. for traceroute/atomiccounter
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## modifysession API

To modify an endpoint (Ep) to Layer 3 (L3) external troubleshooting session, use the **modifysession** API. The module name is **troubleshoot.epextutils.epextsession** and the function is **modifySession**. The required argument (**req\_args**) for the modifysession API is **- session** (session name).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax	Description	Optional Arguments (opt_args)	Description
		- srcep	Source endpoint name
		- dstep	Destination endpoint name
		- srcip	Source endpoint IP address
		- dstip	Destination endpoint IP address

- srcmac	Source endpoint MAC
- dstmac	Destination endpoint MAC
- srcextip	L3 external source IP address
- dstextip	L3 external destination IP address
- starttime	Start time of the troubleshooting session
- endtime	End time of the troubleshooting session
- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
- description	Description about the session
- scheduler	Scheduler name for report generation
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	Format of report to be generated
- ui	Used internally (ignore)
- sessionurl	Location of the report
-action	Start/stop/status etc. for traceroute/atomiccounter
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## atomiccounter API

To create an endpoint (ep) to endpoint atomic counter session, use the **atomiccounter** API. The module name is **troubleshoot.epextutils.epext\_ac** and the function is **manageAtomicCounterPols**.

The required arguments (**req\_args**) for the atomiccounter API include:

- - session (session name)
- - action (start/stop/status)

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name

- dstep	Destination endpoint name
- srcip	Source endpoint IP address
- dstip	Destination endpoint IP address
- srcmac	Source endpoint MAC
- dstmac	Destination endpoint MAC
- srcextip	L3 external source IP address
- dstextip	L3 external destination IP address
- starttime	Start time of the troubleshooting session
- endtime	End time of the troubleshooting session
- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
- ui	Used internally (ignore)
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## traceroute API

To create an endpoint (ep) to Layer 3 external traceroute troubleshooting session using the API, use the **traceroute** API. The module name is **troubleshoot.epextutils.epext\_traceroute** and the function is **manageTraceroutePols**.

The required arguments (**req\_args**) for the traceroute API include:

- - session (session name)
- - action (start/stop/status)

Syntax Description	Optional Arguments (opt_args)	Description
	- protocol	Protocol name
	- dstport	Destination port name
	- srcep	Source endpoint
	- dstep	Destination endpoint
	- srcip	Source IP address
	- dstip	Destination IP address

- srcextip	Source external IP address
- dstIp	Destination external IP address
- ui	Used internally (ignore)
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## span API

To create an endpoint (Ep) to Layer 3 (L3) external span troubleshooting session, use the **span** API. The module name is **troubleshoot.epextutils.epext\_span** and the function is **monitor**.

The required arguments (**req\_args**) for the span API include:

- - session (session name)
- - action (start/stop/status)
- - mode

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- portslst	List of ports
	- dstapic	Destination APIC
	- srcipprefix	Source endpoint IP address prefix
	- flowid	Flow ID
	- dstepg	Destination endpoint group
	- dstip	Destination endpoint IP address
	- analyser	???
	- desttype	Destination type
	- spansrports	Span source ports

## generatereport API

To generate a troubleshooting report using the API, use the **generatereport** API. The module name is **troubleshoot.eptoeputils.report** and the function is **generateReport**.

The required argument (**req\_args**) for the generatereport API is **- session** (session name).



The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name
	- dstep	Destination endpoint name
	- srcip	Source endpoint IP address
	- dstip	Destination endpoint IP address
	- srcmac	Source endpoint MAC
	- dstmac	Destination endpoint MAC
	- srcextip	L3 external source IP address
	- dstextip	L3 external destination IP address
	- starttime	Start time of the troubleshooting session
	- endtime	End time of the troubleshooting session
	- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
	- description	Description about the session
	- scheduler	Scheduler name for report generation
	- srcepid	Obsolete
	- dstepid	Obsolete
	- include	Obsolete
	- format	Format of report to be generated
	- ui	Used internally (ignore)
	- sessionurl	Location of the report
	-action	Start/stop/status etc. for traceroute/atomiccounter
	- mode	Used internally
	- _dc	Used internally
	- ctx	Used internally

## schedulingreport API

To schedule the generation of a troubleshooting report using the API, use the **schedulingreport** API. The module name is **troubleshoot.eptoeptutils.report** and the function is **scheduleReport**. The required argument (**req\_args**) for the schedulingreport API is - **session**

The required arguments (**req\_args**) for the schedulingreport API include:

- - session (session name)
- - scheduler (scheduler name)

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax	Description	Optional Arguments (opt_args)	Description
		- srcep	Source endpoint
		- dstep	Destination endpoint
		- srcip	Source endpoint IP address
		- dstip	Destination endpoint IP address
		- srcmac	Source endpoint MAC
		- dstmac	Destination endpoint MAC
		- srcextip	L3 external source IP address
		- dstextip	L3 external destination IP address
		- starttime	Start time of the troubleshooting session
		- endtime	End time of the troubleshooting session
		- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
		- description	Description about the session
		- srcepid	Obsolete
		- dstepid	Obsolete
		- include	Obsolete
		- format	Format of report to be generated
		- ui	Used internally (ignore)
		- sessionurl	Location of the report
		-action	Start/stop/status etc. for traceroute/atomiccounter
		- mode	Used internally

- _dc	Used internally
- ctx	Used internally

## getreportstatus API

To get the status of a generated report using the API, use the **getreportstatus** API. The module name is **troubleshoot.eptoeputils.report** and the function is **getStatus**.

The required arguments (**req\_args**) for the getreportstatus API include:

- - session (session name)
- - sessionurl (session URL)
- - mode



**Note** There are no optional arguments (**opt\_args**) for the getreportstatus API.

## getreportslist API

To get a list of generated reports using the API, use the **getreportslist** API. The module name is **troubleshoot.eptoeputils.report** and the function is **getReportsList**.

The required arguments (**req\_args**) for the getreportslist API are- **session** (session name) and - **mode**.



**Note** There are no optional arguments (**opt\_args**) for the getreportslist API.

## getsessionslist API

To get a list of troubleshooting sessions using the API, use the **getsessionslist** API. The module name is **troubleshoot.epextutils.epextsession** and the function is **getSessions**.



**Note** There are no required arguments for this API.

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- session	Session name
	- srcep	Source endpoint name
	- dstep	Destination endpoint name

- srcip	Source endpoint IP address
- dstip	Destination endpoint IP address
- srcmac	Source endpoint MAC
- dstmac	Destination endpoint MAC
- srcextip	L3 external source IP address
- dstextip	L3 external destination IP address
- starttime	Start time of the troubleshooting session
- endtime	End time of the troubleshooting session
- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
- description	Description about the session
- scheduler	Scheduler name for report generation
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	Format of report to be generated
- ui	Used internally (ignore)
- sessionurl	Location of report
- action	Start/stop/status etc. for traceroute/atomiccounter
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## getsessiondetail API

To get specific details about a troubleshooting session using the API, use the **getsessiondetail** API. The module name is **troubleshoot.epextutils.session** and the function is **getSessionDetail**. The required argument (**req\_args**) for the **getsessiondetail** API is **- session** (session name).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name

- dstep	Destination endpoint name
- srcip	Source endpoint IP address
- dstip	Destination endpoint IP address
- srcmac	Source endpoint MAC
- dstmac	Destination endpoint MAC
- srcextip	L3 external source IP address
- dstextip	L3 external destination IP address
- starttime	Start time of the troubleshooting session
- endtime	End time of the troubleshooting session
- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
- description	Description about the session
- scheduler	Scheduler name for report generation
- srcepid	Obsolete
- dstepid	Obsolete
- include	Obsolete
- format	Format of report to be generated
- ui	Used internally (ignore)
- sessionurl	Location of report
- action	Start/stop/status etc. for traceroute/atomiccounter
- mode	Used internally
- _dc	Used internally
- ctx	Used internally

## deletesession API

To delete a particular troubleshooting session using the API, use the **deletesession** API. The module name is **troubleshoot.epextutils.epextsession** and the function is **deleteSession**.

The required arguments (**req\_args**) for the deletesession API are **- session** (session name) and **- mode**.



**Note** There are no optional arguments (**opt\_args**) for the deletesession API.

## clearreports API

To clear the list of generated reports using the API, use the **clearreports** API. The module name is **troubleshoot.epextutils.report** and the function is **clearReports**.

The required arguments (**req\_args**) for the clearreports API are- **session** (session name) and - **mode**.



**Note** There are no optional arguments (**opt\_args**) for the clearreports API.

## contracts API

To get contracts information using the API, use the **contracts** API. The module name is **troubleshoot.epextutils.epext\_contracts** and the function is **getContracts**. The required argument (**req\_args**) for the contracts API is - **session** (session name).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax	Description	Optional Arguments (opt_args)	Description
		- srcep	Source endpoint name
		- dstep	Destination endpoint name
		- srcip	Source endpoint IP address
		- dstip	Destination endpoint IP address
		- srcmac	Source endpoint MAC
		- dstmac	Destination endpoint MAC
		- srcextip	L3 external source IP address
		- dstextip	L3 external destination IP address
		- starttime	Start time of the troubleshooting session
		- endtime	End time of the troubleshooting session
		- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
		- epext	Endpoint to external
		- mode	Used internally
		- _dc	Used internally
		- ctx	Used internally
		- ui	Used internally (ignore)

## ratelimit API

This section provides information on the the **ratelimit** API. The module name is **troubleshoot.eptoeputils.ratelimit** and the function is **control**. The required argument (**req\_args**) for the ratelimit API is - **action** (start/stop/status).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name
	- dstep	Destination endpoint name
	- srcip	Source endpoint IP address
	- dstip	Destination endpoint IP address
	- srcmac	Source endpoint MAC
	- dstmac	Destination endpoint MAC
	- srcextip	L3 external source IP address
	- dstextip	L3 external destination IP address
	- starttime	Start time of the troubleshooting session
	- endtime	End time of the troubleshooting session
	- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
	- epext	Endpoint to external
	- mode	Used internally
	- _dc	Used internally
	- ctx	Used internally

## 13ext API

This section provides information on the the **13ext** API. The module name is **troubleshoot.epextutils.13ext** and the function is **execute**. The required argument (**req\_args**) for the 13ext API is - **action** (start/stop/status).

The following table lists the optional arguments (**opt\_args**) and descriptions for each.

Syntax Description	Optional Arguments (opt_args)	Description
	- srcep	Source endpoint name
	- dstep	Destination endpoint name
	- srcip	Source endpoint IP address

- dstip	Destination endpoint IP address
- srcmac	Source endpoint MAC
- dstmac	Destination endpoint MAC
- srcextip	L3 external source IP address
- dstextip	L3 external destination IP address
- starttime	Start time of the troubleshooting session
- endtime	End time of the troubleshooting session
- latestmin	Time window for the troubleshooting session starting from start time (in minutes)
- epext	Endpoint to external
- mode	Used internally

## Checking for Configuration Synchronization Issues

When you make a request in Cisco Application Centric Infrastructure (APIC)—for example, changing a configuration—you generally see immediately that the change has occurred. However, if you encounter an issue with Cisco APIC, you can check in the GUI to see if there are any transactions involving user-configurable objects that have yet to take effect. You can use information in the panel to help with debugging.

The **Configuration Objects Pending Resolution** panel in the Cisco APIC GUI tells you if there are delays.

### Before you begin

- 
- Step 1** Log in to Cisco APIC.
  - Step 2** Click the settings icon (the gear symbol) in the upper right of the screen and choose **Config Sync Issues**.
  - Step 3** In the **Configuration Objects Pending Resolution** panel, check if anything is listed in the table.  
If there are no entries in the table, there are no synchronization issues.
  - Step 4** If there are any entries, capture the information in the table and use it for debugging or working with Cisco support.
- 

## Viewing User Activities

In situations where an admin notices a change to the Cisco APIC setup, the admin can use the **User Activities** feature to view a 2-week history of actions performed by a user. The historical data includes a timestamp of when the action occurred, the user who performed the action, the action the user performed, the affected object, and a description.



## Accessing User Activities

The **User Activities** window enables you to view a 2-week history of user activities performed in the Cisco APIC GUI.

---

**Step 1** From the menu bar, choose **System > Active Sessions** .

The **Active Session** window appears.

**Step 2** Right-click on an active session and choose **User Activities**.

A list of user activities appears.

**Note** For an explanation of a field, click the help icon in the top-right corner of the **Active Session** window to display the help file.

**Step 3** Click the **Actions in the last** drop-down menu to choose how far back in history you want to view the user activities.

---

## Embedded Logic Analyzer Module

### About the Embedded Logic Analyzer Module

ELAM (Embedded Logic Analyzer Module) is an engineering tool that enables you to look inside Cisco ASICs and understand how a packet is being forwarded. ELAM is embedded within the forwarding pipeline and can capture a packet in real time without affecting performance or control plane resources. ELAM can perform the following functions:

- Determine if a packet reached the forwarding engine
- Specify the port and VLAN of the packet that was received
- View the packet (Layer 2 to Layer 4 data)
- Check if the packet was altered where it was sent

### Generating an ELAM Report in the Simplified Output for Modular Switches

The Cisco Application Policy Infrastructure Controller (APIC) 4.2(1) release introduces simplified, human-readable ELAM output. Only switch models with EX, FX, or FX2 at the end of the switch name support the simplified output. Use the following procedure for modular switches.

---

**Step 1** Run the ELAM tool to collect the packet forwarding information. The exact commands and parameters depend on your hardware.

**Step 2** Run the **ereport** command to create ELAM reports of the packet forwarding information in the original format and the simplified format.

**Example:**

```

module-1 (DBG-elam-el6) # ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT

=====
 Trigger/Basic Information
=====
ELAM Report File : /tmp/logs/elam_2019-09-04-51m-13h-30s.txt
.
.
.

module-1 (DBG-elam-el6) # exit
module-1 (DBG-elam) # exit
module-1 # exit

apic1-leaf11# cd /tmp/logs
apic1-leaf11# ls | grep elam
elam_2019-09-04-51m-13h-30s.txt
pretty_elam_2019-09-04-51m-13h-30s.txt
apic1-leaf11#

```

ELAM saves the output files in the `/tmp/logs/` directory. In the example, the `elam_2019-09-04-51m-13h-30s.txt` file is the ELAM report in the original format. The `pretty_elam_2019-09-04-51m-13h-30s.txt` file is the ELAM report in the simplified format. However, the simplified format file will be empty. You must perform additional steps to get the report in the simplified format.

**Step 3** Upload the original format ELAM report to the `/bootflash` directory on the supervisor.

In the example, this report is the `elam_2019-09-04-51m-13h-30s.txt` file.

**Step 4** Log in to the supervisor as admin.

**Step 5** Change the directory to `/tmp`, or any directory with write privileges for the admin user.

**Example:**

```
cd /tmp
```

**Step 6** Run the `decode_elam_parser` command on the original format ELAM report.

**Example:**

```
decode_elam_parser /bootflash/elam_2019-09-04-51m-13h-30s.txt
```

The `decode_elam_parser` command saves the simplified output file in the current directory.

## Generating an ELAM Report in the Simplified Output for Fixed Form-Factor Switches

The Cisco Application Policy Infrastructure Controller (APIC) 4.2(1) release introduces simplified, human-readable ELAM output. Only switch models with EX, FX, or FX2 at the end of the switch name support the simplified output. Use the following procedure for fixed form-factor leaf switches and spine switches.

**Step 1** Run the ELAM tool to collect the packet forwarding information. The exact commands and parameters depend on your hardware.

**Step 2** Run the **ereport** command to create ELAM reports of the packet forwarding information in the original format and the simplified format.

**Example:**

```
module-1(DBG-elam-insel6)# ereport
Python available. Continue ELAM decode with LC Pkg
ELAM REPORT

=====
 Trigger/Basic Information
=====
ELAM Report File : /tmp/logs/elam_2019-09-04-51m-13h-30s.txt
.
.
.

module-1(DBG-elam-insel6)# exit
module-1(DBG-elam)# exit
module-1# exit

apicl-leaf11# cd /tmp/logs
apicl-leaf11# ls | grep elam
elam_2019-09-04-51m-13h-30s.txt
pretty_elam_2019-09-04-51m-13h-30s.txt
apicl-leaf11#
```

ELAM saves the output files in the `/tmp/logs/` directory. In the example, the `elam_2019-09-04-51m-13h-30s.txt` file is the ELAM report in the original format. The `pretty_elam_2019-09-04-51m-13h-30s.txt` file is the ELAM report in the simplified format.

---





## CHAPTER 7

# Troubleshooting Switch Discovery

- [Troubleshooting Switch Discovery Issues, on page 127](#)

## Troubleshooting Switch Discovery Issues

The ACI-mode switch software includes a comprehensive leaf and spine switch discovery validation program. The validation program is launched with a switch CLI command when a switch is stuck in the discovery mode.

The validation program performs the following tests:

1. System state—Checks the state of the `topSystem` managed object (MO).
  - a. If the state is "out-of-service," checks for any scheduled upgrades.
  - b. If the state is "downloading bootscript," a failure has occurred in the downloading bootscript. The failure is reported. If the switch is an L3out spine, the program additionally checks the bootstrap download state and reports any failure.
2. DHCP status—Checks for DHCP status and information, such as the TEP IP, node Id, and name assigned from the `dhcpResp` MO.
3. AV details—Checks whether the APICs are registered and whether they have valid IP addresses.
4. IP reachability—Uses the `iping` command to verify IP reachability to the address assigner APIC. To retest this condition, use the `show discoveryissues apic ipaddress` command.
5. infra VLAN received—Checks for the presence of the infra VLAN details in the `lldpInst` MO. If this switch belongs to a pod that has no APIC, no infra VLAN details are present, and this section of the test result can be ignored.
6. LLDP adjacency—Checks for the presence of LLDP adjacencies and for any wiring mismatch issues. LLDP issues can generate fault reports such as infra VLAN mismatch, chassis ID mismatch, or no connection to the front end ports.
7. Switch version—Reports the running firmware version of the switch. Also reports the version of the APIC, if available.
8. FPGA/BIOS—Checks for any FPGA/BIOS version mismatch on the switch.
9. SSL validation—Checks for validity of the SSL certificate details using the `acdiag verifyssl -s serialNumber` command.

10. Policy downloads—Checks the `pconsBootStrap` MO to see whether registration to APIC (PM shards) is complete and whether all policies were downloaded successfully.
11. Time—Reports the current time on the switch.
12. Hardware status—Checks the module, power, and fan status from the `eqptCh`, `eqptFan`, `eqptPsu`, `eqptFt` and `eqptLC` MOs.

### Running the Test Manually

To run the switch discovery validation program, log in to the spine or leaf switch CLI console and execute the following command:

```
show discoveryissues [apic ipaddress]
```

### Example of a Successful Test

The following example shows the switch discovery validation program output for a successful test.

```
spine1# show discoveryissues

Checking the platform type.....SPINE!
Check01 - System state - in-service [ok]
Check02 - DHCP status [ok]
 TEP IP: 10.0.40.65 Node Id: 106 Name: spine1
Check03 - AV details check [ok]
Check04 - IP rechability to apic [ok]
 Ping from switch to 10.0.0.1 passed
Check05 - infra VLAN received [ok]
 infra vLAN:1093
Check06 - LLDP Adjacency [ok]
 Found adjacency with LEAF
Check07 - Switch version [ok]
 version: n9000-14.2(0.167) and apic version: 5.0(0.25)
Check08 - FPGA/BIOS out of sync test [ok]
Check09 - SSL check [check]
 SSL certificate details are valid
Check10 - Downloading policies [ok]
Check11 - Checking time [ok]
 2019-08-21 17:15:45
Check12 - Checking modules, power and fans [ok]
```

### Example of a Failed Test

The following example shows the switch discovery validation program output for a switch with discovery issues.

```
spine1# show discoveryissues

Checking the platform type.....SPINE!
Check01 - System state - out-of-service [FAIL]
 Upgrade status is notscheduled
 Node upgrade is notscheduled state
Check02 - DHCP status [FAIL]
 ERROR: discover not being sent by switch
 Ignore this, if the IP is already known by switch
 ERROR: node Id not configured
 ERROR: Ip not assigned by dhcp server
```

```
ERROR: Address assigner's IP not populated
TEP IP: unknown Node Id: unknown Name: unknown
Check03 - AV details check [ok]
Check04 - IP reachability to apic [FAIL]
 please rerun the CLI with argument apic Ip
 (show discoveryissues apic <ip>) to check its reachability from switch
Check05 - infra VLAN received [FAIL]
 Please ignore if this switch is part of a pod with no apic
Check06 - LLDP Adjacency [FAIL]
 Error: spine not connected to any leaf
Check07 - Switch version [ok]
 version: n9000-14.2(0.146) and apic version: unknown
Check08 - FPGA/BIOS out of sync test [ok]
Check09 - SSL check [ok]
 SSL certificate details are valid
Check10 - Downloading policies [FAIL]
 Registration to all PM shards is not complete
 Policy download is not complete
 Pcons bootstrap is in triggered state
Check11 - Checking time [ok]
 2019-07-17 19:26:29
Check12 - Checking modules, power and fans [FAIL]
 Line card state is testing
```







## CHAPTER 8

# Manually Removing Disabled Interfaces and Decommissioned Switches from the GUI

---

In a scenario where a fabric port is shut down then brought back up, it is possible that the port entry will remain disabled in the GUI. If this occurs, no operations can be performed on the port. To resolve this, the port must be manually removed from the GUI.

- [Manually Removing Disabled Interfaces and Decommissioned Switches from the GUI, on page 131](#)

## Manually Removing Disabled Interfaces and Decommissioned Switches from the GUI

This section explains how to manually remove disabled interfaces and decommissioned switches in the GUI.

- 
- Step 1** From the **Fabric** tab, click **Inventory**.
  - Step 2** In the **Navigation** pane, click **Disabled Interfaces and Decommissioned Switches**.  
The list of disabled interfaces and decommissioned switches appears in a summary table in the **Work** pane.
  - Step 3** From the **Work** pane, right-click on the interface or switch that you want to remove and choose **Delete**.
-





## CHAPTER 9

# Decommissioning and Recommissioning Switches

---

This chapter contains the following sections:

- [Decommissioning and Recommissioning Switches, on page 133](#)

## Decommissioning and Recommissioning Switches

To decommission and recommission all the nodes in a pod, perform this procedure. One use case for this is to change the node IDs to a more logical, scalable numbering convention.

---

**Step 1** Decommission the nodes in the pod by following these steps for each one:

- Navigate to **Fabric > Inventory** and expand the **Pod**.
- Select the switch, right-click on it, and choose **Remove from Controller**.
- Confirm the action and click **OK**.

The process takes about 10 minutes. The node is automatically wiped and reloaded. In addition, the node configuration is removed from the controller.

- If a decommissioned node had the port profile feature deployed on it, some port configurations are not removed with the rest of the configuration. It is necessary to manually delete the configurations after the decommission for the ports to return to the default state. To do this, log on to the switch, run the **setup-clean-config.sh** script, and wait for it to run. Then, enter the **reload** command.

**Step 2** When all the switches have been decommissioned from the pod, verify they are all physically connected and booted in the desired configuration.

**Step 3** Perform the following actions to recommission each node.

**Note** Before recommissioning a node with a port profile configuration as a new node, you must run the **setup-clean-config.sh** script to restore the port configuration to the default settings.

- Navigate to **Fabric > Inventory**, expand **Quick Start**, and click **Node or Pod Setup**.
- Click **Setup Node**.
- In the **Pod ID** field, choose the pod ID.
- Click the + to open the **Nodes** table.
- Enter the node ID, serial number, Switch name, TEP Pool ID, and Role (**leaf** or **spine**) for the switch.

f) Click **Update**.

**Step 4** Verify the nodes are all set up by navigating to **Fabric > Inventory > Fabric Membership**.

---

#### **What to do next**

If the pod is one of the pods in a multipod topology, reconfigure multipod for this pod and the nodes. For more information, see *Multipod* in the *Cisco APIC Layer 3 Networking Configuration Guide*.



## CHAPTER 10

# Troubleshooting Steps for Endpoint Connectivity Problems

---

This chapter lists the steps for troubleshooting endpoint connectivity issues using the Cisco APIC tools, contains procedures for inspecting the operational status of your endpoints and tunnel interfaces, and explains how to connect an SFP module.

This chapter contains the following sections:

- [Troubleshooting Endpoint Connectivity, on page 135](#)
- [Inspecting Endpoint and Tunnel Interface Status, on page 136](#)
- [Connecting an SFP Module, on page 137](#)

## Troubleshooting Endpoint Connectivity

---

**Step 1** Inspect the operational status of each endpoint.

The operational status will reveal any fault or misconfiguration of the endpoints. See [Inspecting the Endpoint Status, on page 136](#).

**Step 2** Inspect the status of the tunnel interface.

The operational status will reveal any fault or misconfiguration of the tunnel. See [Inspecting the Tunnel Interface Status, on page 136](#).

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

A traceroute will reveal any problems with intermediate nodes, such as spine nodes, between the endpoints. See [Performing a Traceroute Between Endpoints, on page 88](#).

**Step 4** Configure an atomic counter on an endpoint.

The atomic counter will confirm whether the source endpoint is transmitting packets or the destination endpoint is receiving packets, and whether the number of packets received equals the number of packets sent. See [Configuring Atomic Counters, on page 34](#).

**Step 5** Inspect the contracts under each EPG.

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

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

A packet analyzer on the monitoring node will reveal any packet issues such as an incorrect address or protocol. See [Configuring a Tenant SPAN Session Using the Cisco APIC GUI, on page 54](#).

## Inspecting Endpoint and Tunnel Interface Status

This section explains how to inspect the operational status of endpoints and tunnel interfaces. Performing these procedures enables you to reveal any fault or misconfiguration of the endpoints and tunnel interfaces.

### Inspecting the Endpoint Status

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

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

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

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

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

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

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

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

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

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

**Step 10** Repeat this procedure for the destination endpoint.

**Note** Occasionally, bidirectional traffic is interrupted between IP addresses in two micro-segmented EPGs deployed behind two leaf switches in the fabric. This can occur when the IP addresses are transitioning because of a configuration change from micro-segment EPG to base EPG. Or conversely, this can occur on two different leaf switches at the same time while bidirectional traffic is running. In this case, the policy tag for each remote endpoint still points to its previous EPG.

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

### Inspecting the Tunnel Interface Status

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

- 
- Step 1** In the menu bar, click **Fabric**.
- Step 2** In the submenu bar, click **Inventory**.
- Step 3** In the **Navigation** pane, expand the pod and expand the node ID of the source endpoint interface.
- Step 4** Under the node, expand **Interfaces**, expand **Tunnel Interfaces**, and click the tunnel ID of the source endpoint interface.
- Step 5** In the **Work** pane, verify the tunnel interface properties and click the **Operational** tab.
- Step 6** In the **Operational** tab, view the health, status, and fault information.  
In the **Status** table, click any items with entries, such as changes, events, or faults.
- Step 7** Repeat this procedure for the destination endpoint interface.
- 

## Connecting an SFP Module

When you connect an SFP module to a new card, you need to create a link speed policy for the module to communicate with the card. Follow these steps to create a link speed policy.

- 
- Step 1** Create an interface policy to specify the link speed:

**Example:**

```
<fabricHIIfPol name="SpeedPol" speed="1G"/>
```

- Step 2** Reference the link speed policy within an interface policy group:

**Example:**

```
<infraAccPortGrp name="myGroup">
 <infraRsHIIfPol tnFabricHIIfPolName="SpeedPol"/>
</infraAccPortGrp>
```

---







# CHAPTER 11

## Troubleshooting EVPN Type-2 Route Advertisement

- [Troubleshooting EVPN Type-2 Route Distribution to a DCIG, on page 139](#)

### Troubleshooting EVPN Type-2 Route Distribution to a DCIG

For optimal traffic forwarding in an EVPN topology, you can enable fabric spines to distribute host routes to a Data Center Interconnect Gateway (DCIG) using EVPN type-2 (MAC-IP) routes along with the public BD subnets in the form of BGP EVPN type-5 (IP Prefix) routes. This is enabled using the HostLeak object. If you encounter problems with route distribution, use the steps in this topic to troubleshoot.

#### SUMMARY STEPS

1. Verify that HostLeak object is enabled under the VRF-AF in question, by entering a command such as the following in the spine-switch CLI:
2. Verify that the config-MO has been successfully processed by BGP, by entering a command such as the following in the spine-switch CLI:
3. Verify that the public BD-subnet has been advertised to DCIG as an EVPN type-5 route:
4. Verify whether the host route advertised to the EVPN peer was an EVPN type-2 MAC-IP route:
5. Verify that the EVPN peer (a DCIG) received the correct type-2 MAC-IP route and the host route was successfully imported into the given VRF, by entering a command such as the following on the DCIG device (assuming that the DCIG is a Cisco ASR 9000 switch in the example below):

#### DETAILED STEPS

**Step 1** Verify that HostLeak object is enabled under the VRF-AF in question, by entering a command such as the following in the spine-switch CLI:

**Example:**

```
spine1# ls /mit/sys/bgp/inst/dom-apple/af-ipv4-ucast/
ctrl-l2vpn-evpn ctrl-vpnv4-ucast hostleak summary
```

**Step 2** Verify that the config-MO has been successfully processed by BGP, by entering a command such as the following in the spine-switch CLI:

**Example:**

```
spine1# show bgp process vrf apple
```

Look for output similar to the following:

```
Information for address family IPv4 Unicast in VRF apple
Table Id : 0
Table state : UP
Table refcount : 3
Peers : 0
Active-peers : 0
Routes : 0
Paths : 0
Networks : 0
Aggregates : 0

Redistribution
None

Wait for IGP convergence is not configured
GOLF EVPN MAC-IP route is enabled
EVPN network next-hop 192.41.1.1
EVPN network route-map map_pfxleakctrl_v4
Import route-map rtctrlmap-apple-v4
EVPN import route-map rtctrlmap-evpn-apple-v4
```

**Step 3** Verify that the public BD-subnet has been advertised to DCIG as an EVPN type-5 route:

**Example:**

```
spine1# show bgp l2vpn evpn 10.6.0.0 vrf overlay-1
Route Distinguisher: 192.41.1.5:4123 (L3VNI 2097154)
BGP routing table entry for [5]:[0]:[0]:[16]:[10.6.0.0]:[0.0.0.0]/224, version 2088
Paths: (1 available, best #1)
Flags: (0x000002 00000000) on xmit-list, is not in rib/evpn
Multipath: eBGP iBGP

Advertised path-id 1
Path type: local 0x4000008c 0x0 ref 1, path is valid, is best path
AS-Path: NONE, path locally originated
192.41.1.1 (metric 0) from 0.0.0.0 (192.41.1.5)
Origin IGP, MED not set, localpref 100, weight 32768
Received label 2097154
Community: 1234:444
Extcommunity:
RT:1234:5101
4BYTEAS-GENERIC:T:1234:444

Path-id 1 advertised to peers:
50.41.50.1
```

In the **Path type** entry, **ref 1** indicates that one route was sent.

**Step 4** Verify whether the host route advertised to the EVPN peer was an EVPN type-2 MAC-IP route:

**Example:**

```
spine1# show bgp l2vpn evpn 10.6.41.1 vrf overlay-1
Route Distinguisher: 10.10.41.2:100 (L2VNI 100)
BGP routing table entry for [2]:[0]:[2097154]:[48]:[0200.0000.0002]:[32]:[10.6.41.1]/272, version 1146
Shared RD: 192.41.1.5:4123 (L3VNI 2097154)
Paths: (1 available, best #1)
Flags: (0x00010a 00000000) on xmit-list, is not in rib/evpn
Multipath: eBGP iBGP

Advertised path-id 1
Path type: local 0x4000008c 0x0 ref 0, path is valid, is best path
```

```
AS-Path: NONE, path locally originated
EVPN network: [5]:[0]:[0]:[16]:[10.6.0.0]:[0.0.0.0] (VRF apple)
 10.10.41.2 (metric 0) from 0.0.0.0 (192.41.1.5)
 Origin IGP, MED not set, localpref 100, weight 32768
 Received label 2097154 2097154
 Extcommunity:
 RT:1234:16777216
```

```
Path-id 1 advertised to peers:
 50.41.50.1
```

The **Shared RD** line indicates the RD/VNI shared by the EVPN type-2 route and the BD subnet.

The **EVPN Network** line shows the EVPN type-5 route of the BD-Subnet.

The **Path-id advertised to peers** indicates the path advertised to EVPN peers.

**Step 5**

Verify that the EVPN peer (a DCIG) received the correct type-2 MAC-IP route and the host route was successfully imported into the given VRF, by entering a command such as the following on the DCIG device (assuming that the DCIG is a Cisco ASR 9000 switch in the example below):

**Example:**

```
RP/0/RSP0/CPU0:asr9k#show bgp vrf apple-2887482362-8-1 10.6.41.1
Tue Sep 6 23:38:50.034 UTC
BGP routing table entry for 10.6.41.1/32, Route Distinguisher: 44.55.66.77:51
Versions:
 Process bRIB/RIB SendTblVer
 Speaker 2088 2088
Last Modified: Feb 21 08:30:36.850 for 28w2d
Paths: (1 available, best #1)
 Not advertised to any peer
 Path #1: Received by speaker 0
 Not advertised to any peer
Local
 192.41.1.1 (metric 42) from 10.10.41.1 (192.41.1.5)
 Received Label 2097154
 Origin IGP, localpref 100, valid, internal, best, group-best, import-candidate, imported
 Received Path ID 0, Local Path ID 1, version 2088
 Community: 1234:444
 Extended community: 0x0204:1234:444 Encapsulation Type:8 Router
MAC:0200.c029.0101 RT:1234:5101
 RIB RNH: table_id 0xe0000190, Encap 8, VNI 2097154, MAC Address: 0200.c029.0101,
 IP Address: 192.41.1.1, IP table_id 0x00000000
 Source AFI: L2VPN EVPN, Source VRF: default,
 Source Route Distinguisher: 192.41.1.5:4123
```

In this output, the received RD, next hop, and attributes are the same for the type-2 route and the BD subnet.





## CHAPTER 12

# Performing a Rebuild of the Fabric

This chapter explains how to rebuild your fabric.

- [Rebuilding the Fabric, on page 143](#)

## Rebuilding the Fabric



---

**Caution** This procedure is extremely disruptive. It eliminates the existing fabric and recreates a new one.

---

This procedure allows you to rebuild (reinitialize) your fabric, which you may need to do for any of the following reasons:

- To change the TEP IPs
- To change the Infra VLAN
- To change the fabric name
- To perform TAC troubleshooting tasks

Deleting the APICs erases the configuration on them and brings them up in the startup script. Performing this on the APICs can be done in any order, but ensure that you perform the procedure on all of them (every leaf and spine in the fabric).

### Before you begin

Ensure that the following is in place:

- Regularly scheduled backups of the configuration
- Console access to the leaves and spines
- A configured and reachable CIMC, which is necessary for KVM console access
- No Java issues

- 
- Step 1** If you would like to retain your current configuration, you can perform a configuration export. For more information, see the *Cisco ACI Configuration Files: Import and Export* document: <https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>
- Step 2** Erase the configuration on the APICs by connecting to the KVM console and entering the following commands:
- >acidiag touch clean**
  - >acidiag touch setup**
  - >acidiag reboot**
- Ensure that each node boots up in fabric discovery mode and is not part of the previously configured fabric.
- Note** The **acidiag touch** command alone is not useful for this procedure, because it does not bring the APIC up in the startup script.
- Caution** It is extremely important that you ensure that all previous fabric configurations have been removed. If any previous fabric configuration exists on even a single node, the fabric cannot be rebuilt.
- Step 3** When all previous configurations have been removed, run the startup script for all APICs. At this point, you can change any of the above values, TEP, TEP Vlan, and/or Fabric Name. Ensure that these are consistent across all APICs. For more information, refer to the *Cisco APIC Getting Started Guide*: <https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>.
- Step 4** To clean reboot the fabric nodes, log in to each fabric node and execute the following:
- >setup-clean-config.sh**
  - >reload**
- Step 5** Log in to apic1 and perform a configuration import. For more information, see the *Cisco ACI Configuration Files: Import and Export* document: <https://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html>.
- Step 6** Wait for a few minutes as the fabric now uses the previous fabric registration policies to rebuild the fabric over the nodes. (Depending on the size of the fabric, this step may take awhile.)
-



## CHAPTER 13

# Verifying IP-Based EPG Configurations

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

This chapter contains the following sections:

- [Verifying IP-Based EPG Configurations Using the GUI, on page 145](#)
- [Verifying IP-EPG Configurations Using Switch Commands, on page 146](#)

## Verifying IP-Based EPG Configurations Using the GUI

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

- 
- Step 1** Verify that the IP-based EPG you created is listed under the **uSeg EPGs** folder in the GUI (shown in the following screen capture).  
Note that there is one IP-based EPG listed under uSeg EPGs named "IP" that was created using the REST API.
- Step 2** Verify that the information is correct in the EPG - IP properties screen (right side window pane) for each EPG IP (IP-based EPG).  
Note the list of IP-based EPGs and IP addresses that are shown at the bottom of the screen.
- Step 3** From your web browser, enter the APIC IP address followed by "/visore.html." Visore is a tool that allows you to view all the objects in the system, such as EPGs. You can use Visore to verify that your IP-based EPGs have been properly configured. For more information about Visore, see the *Application Policy Infrastructure Controller Visore Tool Introduction* document.
- Step 4** Enter your username and password then click **Login** to log into Visore.
- Step 5** Run a query for the IP-based EPGs that you verified in the GUI by entering the name of the class in the field next to **Class or DN** (for example, "fvAEPg").

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

- Step 6** This is a view from the switch point of view. On the switch, you can run a query for the fvEpP class to see the EPGs and check for the "crtrnEnabled" attribute. It will be set to "yes" for IP-based EPGs. Verify that under this EPG, the children of the EPG are shown with IP addresses to ensure a proper configuration. For each IP address configured, there is one object (named "l3IpCktEp") that the switch uses to classify the traffic. Once the configuration is there, when the packets arrive, the switch uses these objects to classify them.
- Step 7** Verify that the pcTags for all the endpoints and IP addresses that you configured match. Every EPG has a pcTag. All the endpoints that match with the IP addresses you configured are classified into this pcTag. Every endpoint has an IP address that you can run a class query on. When you are troubleshooting, you want to verify whether these endpoints (servers) are properly getting classified into this IP-based EPG or not. (The pcTags should match for the IP-based EPG.)

## Verifying IP-EPG Configurations Using Switch Commands

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

- Step 1** Log in to the leaf.
- Step 2** Navigate to the /mit/sys directory.
- Step 3** In the /mit/sys directory, find ctx (vrf context directory)
- Step 4** In the VRF ctx directory, go to the specific BD directory where the IpCkt is configured. You should see the IpCkt.
- Note** "IpCkt" and "IP-EPG" are used interchangeably in this document.
- Step 5** Navigate to the directory and the "cat summary" gives you the information regarding IpCkt.
- Step 6** Ensure that the summary's "operSt" does not say "unsupported".
- Step 7** Find out the VLAN ID that corresponds to the BD where the IpCkt is configured.
- Note** The VLAN ID can be found through any of the **show vlan internal bd-info** commands or through the **show system internal epm vlan all** command.
- Step 8** Once you find the VLAN ID of the BD, issue **show system internal epm <vlan-id> detail**. Here you should be able to see all the configured IpCkts with a specific sclass. (It should match that of what you see in the /mit/sys directory.)
- Step 9** Repeat the steps for vsh\_lc that you followed for vsh.
- Step 10** Send the traffic with an IP matching the IpCkt in the BD, and through **show system internal epm endp ip <a.b.c.d>**, you can verify that the learned IP has the IP-flags for "sclass" and a specific sclass value.
- Step 11** Repeat the steps for vsh\_lc that you followed for vsh.

List of the Switch Troubleshooting Commands Used in this Procedure:

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



```
Vsh_lc -c "show vlan internal bd-info"
Vsh_lc -c "show system internal epm vlan <vlan-id> detail"
vsh_lc -c "show system internal epm endp ip <a.b.c.d>"
vsh_lc -c "show system internal epm epg"
```





# CHAPTER 14

## Recovering a Disconnected Leaf

If all fabric interfaces on a leaf are disabled (interfaces connecting a leaf to the spine) due to a configuration pushed to the leaf, connectivity to the leaf is lost forever and the leaf becomes inactive in the fabric. Trying to push a configuration to the leaf does not work because connectivity has been lost. This chapter describes how to recover a disconnected leaf.

- [Recovering a Disconnected Leaf Using the NX-OS-Style CLI, on page 149](#)
- [Recovering a Disconnected Leaf Using the REST API, on page 150](#)

### Recovering a Disconnected Leaf Using the NX-OS-Style CLI

This procedure enables fabric interfaces using the Cisco Application Policy Infrastructure Controller (APIC) NX-OS-style CLI. Use this procedure if you do not have any external tools from which you can make REST API calls.



**Note** This procedure assumes that 1/31 is one of the leaf switch ports connecting to the spine switch.

**Step 1** Using Cisco APIC NX-OS-style CLI, remove the block list policy.

**Example:**

```
apic1# podId='1'
apic1# nodeId='103'
apic1# interface='eth1/31'
apic1# icurl -sX POST 'http://127.0.0.1:7777/api/mo/.json' -d '{"fabricRsOosPath":{"attributes":
{ "dn": "uni/fabric/outofsvc/rsOosPath-[topology/pod-'$podId']/paths-'$nodeId'/pathep-['$interface']", "status": "deleted" } }'
```

**Step 2** Using the CLI of a leaf or spine switch, set the port in service to bring up the port on the leaf switch.

**Example:**

```
switch1# podId='1'
switch1# nodeId='103'
switch1# interface='eth1/31'
switch1# icurl -X POST
'http://127.0.0.1:7777/api/node/mo/topology/pod-'$podId'/node-'$nodeId'/sys/action.json'
-d
'{"actionLSubj":{"attributes":{"oDn":"sys/phys-['$interface']"},"children":[{"11EthIfSetInServiceLTask":
```

```
{"attributes":{"adminSt":"start"}}}]}}}'
```

## Recovering a Disconnected Leaf Using the REST API

To recover a disconnected leaf switch, you must enable at least one of the fabric interfaces using this procedure. You can enable the remaining interfaces using the GUI, REST API, or CLI.

To enable the first interface, post a policy using the REST API to delete the policy posted and bring the fabric ports Out-of-Service. You can post a policy to the leaf switch to bring the port that is Out-of-Service to In-Service as follows:



**Note** This procedure assumes that 1/49 is one of the leaf switch ports connecting to the spine switch.

**Step 1** Clear the block list policy from the Cisco APIC using the REST API.

**Example:**

```
$APIC_Address/api/policymgr/mo/.xml
<polUni>
 <fabricInst>
 <fabricOOServicePol>
 <fabricRsOosPath tDn="topology/pod-1/paths-$LEAF_Id/pathep-[eth1/49]" lc="blacklist"
status ="deleted"/>
 </fabricOOServicePol>
 </fabricInst>
</polUni>
```

**Step 2** Post a local task to the node itself to bring up the interfaces you want using **l1EthIfSetInServiceLTask**.

**Example:**

```
$LEAF_Address/api/node/mo/topology/pod-1/node-$LEAF_Id/sys/action.xml
<actionLSubj oDn="sys/phys-[eth1/49]">
 <l1EthIfSetInServiceLTask adminSt='start' />
</actionLSubj>
```



## CHAPTER 15

# Troubleshooting a Loopback Failure

- [Identifying a Failed Line Card, on page 151](#)

## Identifying a Failed Line Card

This section explains how to identify a failed line card when getting a loopback failure.

### Before you begin

You should have created a On-Demand TechSupport policy for the fabric node. If you have not already created an On-Demand TechSupport policy, see the “Sending an On-Demand Tech Support File Using the GUI” section in the *Cisco APIC Basic Configuration Guide*.

- 
- Step 1** Collect the Logs Location file of the On-Demand TechSupport policy for the fabric node. To initiate the collection:
- In the menu bar, click **Admin**.
  - In the submenu bar, click **Import/Export**.
  - In the **Navigation** pane, expand **Export Policies** and right-click the On-Demand TechSupport policy for the fabric node.  
A list of options appears.
  - Choose **Collect Tech Supports**.  
The **Collect Tech Supports** dialog box appears.
  - In the **Collect Tech Supports** dialog box, click **Yes** to begin collecting tech support information.
- Step 2** Download the the Logs Location file of the On-Demand TechSupport policy for the fabric node. To download the Logs Location file:
- From the On-Demand TechSupport policy window in the **Work** pane, click the **Operational** tab.  
A summary table appears in the On-Demand TechSupport policy window with several columns, including the **Logs Location** column.
  - Click the URL in the **Logs Location** column.
- Step 3** Inside the Logs Location file, go to the `/var/sysmgr/tmp_logs/` directory and unzip the `svc_ifc_techsup_nxos.tar` file.
- ```
-bash-4.1$ tar xopf svc_ifc_techsup_nxos.tar
```
- The `show_tech_info` directory is created.
- Step 4** Run `zgrep "fcl-cconn failed" show-tech-sup-output.gz | less`.

```
-bash-4.1$ zgrep "fclc-conn failed" show-tech-sup-output.gz | less
[103] diag_port_lb_fail_module: Bringing down the module 25 for Loopback test failed. Packets possibly
lost on the switch SPINE or LC fabric (fclc-conn failed)
[103] diag_port_lb_fail_module: Bringing down the module 24 for Loopback test failed. Packets possibly
lost on the switch SPINE or LC fabric (fclc-conn failed)
```

Note The **fclc-conn failed** message indicates a failed line card.

Step 5 Power cycle the currently failed fabric cards and ensure the fabric cards come online.

Step 6 If the fabric cards fail to come online, or after the fabric cards go offline again, immediately collect the `diag_port_lb.log` file and send the file to the TAC team. The `diag_port_lb.log` file is located in the `/var/sysmgr/tmp_logs/` directory of the Logs Location file.



CHAPTER 16

Determining Why a PIM Interface Was Not Created

A PIM interface (pim:if) is created for L3Out interfaces (note that L3Out SVI interfaces are not supported), multicast tunnel interfaces (per VRF), SVI interfaces corresponding to PIM-enabled pervasive BDs, and loopback interfaces on border leafs (each per VRF).

This chapter contains troubleshooting information for situations where the pim:if is not being created. For more information on PIM, see the *Cisco ACI and Layer 3 Multicast with Cisco ACI* and the *Cisco Application Centric Infrastructure Fundamentals* guides.

This chapter contains the following sections:

- [A PIM Interface Was Not Created For an L3Out Interface, on page 153](#)
- [A PIM Interface Was Not Created For a Multicast Tunnel Interface, on page 154](#)
- [A PIM Interface Was Not Created For a Multicast-Enabled Bridge Domain, on page 154](#)

A PIM Interface Was Not Created For an L3Out Interface

If a PIM interface (pim:If) is not being created for an L3Out interface, confirm the following:

1. PIM is enabled on the L3Out. If PIM is disabled, enable it.
2. If PIM is enabled on the container L3Out, confirm that a multicast l3ext:InstP has been created with "__int_" as a prefixed name. This multicast l3ext:InstP is used to deploy L3Out PIM policies to the switches. There should be one multicast l3ext:InstP per L3Out.



Note

- If a multicast l3ext:InstP exists on the IFC, we can check whether a corresponding fv:RtdEpP is created and deployed on each switch where there is an interface in that L3Out.
 - We do not support an L3Out SVI interface for PIM.
-

A PIM Interface Was Not Created For a Multicast Tunnel Interface

If a PIM interface (pim:if) is not created for a multicast tunnel interface (tunnel:If), confirm the following:

1. The corresponding tunnel:If has been created.



Note The tunnel:If should have type “underlay-mcast.”

2. Each mcast-enabled VRF has created an mcast tunnel.
3. The destination IP field of the tunnel:If is populated with a valid GIPO address.
4. If the tunnel:If is not populated with a valid GIPO address, check the pim:CtxP on the IFC and the pim:CtxDef on the switches to make sure GIPO is allocated correctly.
5. The source IP of the tunnel:If has the loopback address of an L3Out for BL and “127.0.0.100” for NBL.

A PIM Interface Was Not Created For a Multicast-Enabled Bridge Domain

If a PIM interface (pim:if) is not created for a multicast-enabled bridge domain (BD), confirm the following:

1. The corresponding BD or corresponding Ctx has PIM enabled.
2. The corresponding BD is pervasive.
3. The pervasive BD-based pim:If takes default parameters.



Note For interaction with igmp snooping, when PIM is enabled on a pervasive BD, the routing bit should be automatically enabled for the corresponding igmpsnoop:If.



CHAPTER 17

Confirming the Port Security Installation

This chapter explains how to confirm the port security installation in the APIC and leaf switch using Visore and how to confirm port security has been programmed in the hardware using the Cisco NX-OS-style CLI. For information about configuring port security, see the *Cisco Port Security* document.

This chapter contains the following sections:

- [Confirming Your Port Security Installation Using Visore](#) , on page 155
- [Confirming Your Hardware Port Security Installation Using the Cisco NX-OS CLI](#), on page 155

Confirming Your Port Security Installation Using Visore

Step 1 On the Cisco APIC, run a query for the I2PortSecurityPol class in Visore to verify the port security policy installation.

Step 2 On the leaf switch, run a query for I2PortSecurityPolDef in Visore to confirm that the concrete object exists on the interface.

If you have confirmed that port security is installed on the Cisco APIC and leaf switch, use the Cisco NX-OS CLI to confirm that port security has been programmed in the hardware.

Confirming Your Hardware Port Security Installation Using the Cisco NX-OS CLI

Step 1 View the port security status on the switch interface as follows:

Example:

```
switch# show system internal epm interface ethernet 1/35 det
name : Ethernet1/35 ::: if index : 0x1a022000 ::: state : UP
vPC : No ::: EPT : 0x0
MAC Limit : 8 ::: Learn Disable : No ::: PortSecurity Action : Protect
VLANs : 4-23
Endpoint count : 5
Active Endpoint count : 5
```

```
switch# show system internal epm interface port-channel 1 det

name : port-channell1 ::: if index : 0x16000000 ::: state : UP
vPC : No ::: EPT : 0x0
MAC Limit : 6 ::: Learn Disable : No ::: PortSecurity Action : Protect
VLANs :
Endpoint count : 0
Active Endpoint count : 0
Number of member ports : 1
Interface : Ethernet1/34 /0x1a021000
::::
```

Step 2 View the port security status on the module interface as follows:

Example:

```
module-1# show system internal epmc interface ethernet 1/35 det
if index : 0x1a022000 ::: name : Ethernet1/35 ::: tun_ip = 0.0.0.0
MAC limit : 8 ::: is_learn_disable : No ::: MAC limit action: Protect
pc if index : 0 ::: name :
is_vpc_fc FALSE ::: num_mem_ports : 0
interface state : up
Endpoint count : 5
EPT : 0

module-1# show system internal epmc interface port-channel 1 det
if index : 0x16000000 ::: name : port-channell1 ::: tun_ip = 0.0.0.0
MAC limit : 6 ::: is_learn_disable : No ::: MAC limit action: Protect
pc if index : 0 ::: name :
is_vpc_fc FALSE ::: num_mem_ports : 1
interface state : up
Endpoint count : 0
EPT : 0
::::
```

Step 3 View the port security status on the leaf switch as follows:

Example:

```
swtb15-leaf2# show system internal epm interface ethernet 1/35 det

name : Ethernet1/35 ::: if index : 0x1a022000 ::: state : UP
vPC : No ::: EPT : 0x0
MAC Limit : 5 ::: Learn Disable : Yes ::: PortSecurity Action : Protect
VLANs : 4-23
Endpoint count : 5
Active Endpoint count : 5
::::
```

Step 4 Confirm the MAC limit on the module interface as follows:

Example:

```
module-1# show system internal eltmc info interface port-channell1 | grep mac_limit
mac_limit_reached: 0 ::: mac_limit: 8
port_sec_feature_set: 1 ::: mac_limit_action: 1
```

Example:

```
module-1# show system internal eltmc info interface ethernet 1/35 | grep mac_limit
mac_limit_reached: 0 ::: mac_limit: 8
port_sec_feature_set: 1 ::: mac_limit_action: 1
```

Step 5 View the port security status in the module and confirm the MAC limit as follows:

Example:

```
module-1# show system internal epmc interface ethernet 1/35 det
if index : 0x1a022000 ::: name : Ethernet1/35 ::: tun_ip = 0.0.0.0
MAC limit : 5 ::: is_learn_disable : Yes ::: MAC limit action: Protect
pc if index : 0 ::: name :
is_vpc_fc FALSE ::: num_mem_ports : 0
interface state : up
Endpoint count : 5
EPT : 0
::::
```

Example:

```
module-1# show system internal eltmc info interface ethernet 1/35 | grep mac_limit
mac_limit_reached:          1  ::: mac_limit:          5
port_sec_feature_set:      1  ::: mac_limit_action:    1
module-1# exit
```



CHAPTER 18

Troubleshooting QoS Policies

This section provides solutions for troubleshooting QoS policies.

- [Troubleshooting Cisco APIC QoS Policies, on page 159](#)

Troubleshooting Cisco APIC QoS Policies

The following table summarizes common troubleshooting scenarios for Cisco APIC QoS.

| Problem | Solution |
|---|--|
| Unable to update a configured QoS policy. | <ol style="list-style-type: none">1. Invoke the following API to ensure that <code>qospDscpRule</code> is present on the leaf.
<pre>GET https://192.0.20.123/api/node/class/qospDscpRule.xml</pre>2. Ensure that the QoS rules are accurately configured and associated to the EPG ID to which the policy is attached.

Use the following NX-OS style CLI commands to verify the configuration.
<pre>leaf1#
show vlan

leaf1#
show system internal aclqos qos policy detail

apic1#
show running-config tenant <i>tenant-name</i> policy-map type
qos <i>custom-qos-policy-name</i>

apic1#
show running-config tenant <i>tenant-name</i> application
<i>application-name</i> epg <i>epg-name</i></pre> |

| Problem | Solution |
|---------------------------------------|---|
| <p>Show QoS interface statistics.</p> | <p>CLI displays statistics for eth1/1 for only QoS classes – level1, leve2, level3, level4, level5, level6, and policy-plane – if you don’t use “detail” option.</p> <pre data-bbox="599 373 1300 422"> NXOS ibash cli: tor-leaf1# show queuing interface ethernet 1/1 [detail] </pre> <p>If you want to display statistics for control-plane and span classes for an interface, you need to use CLI with the “detail” option.</p> <p>Example: fabric 107 show queuing interface ethernet 1/1 detail</p> <pre data-bbox="599 569 1419 617"> APIC CLI: swtb123-ifc1# fabric node_id show queuing interface ethernet 1/1 </pre> |



CHAPTER 19

Determining the Supported SSL Ciphers

This chapter explains how to determine which SSL ciphers are supported.

- [About SSL Ciphers, on page 161](#)
- [Determining the Supported SSL Ciphers Using the CLI , on page 162](#)

About SSL Ciphers

The Cisco Application Centric Infrastructure (ACI) Representational State Transfer (REST) Application Programming Interface (API) has gone through an evolution from the day the solution debuted to recent versions where the HTTPS/SSL/TLS support has gotten increasingly more stringent. This document is intended to cover the evolution of HTTPS, SSL, and TLS support on the Cisco ACI REST API and provide customers with a guide of what is required for a client to utilize the REST API securely.

HTTPS is a protocol that utilizes either Secure Socket Layers (SSL) or Transport Layer Security (TLS) to form a secure connection for a HTTP session. SSL or TLS is used to encrypt the traffic between a client and a HTTP server. In addition, servers that support HTTPS have a certificate that can usually be used by the client to verify the server's authenticity. This is the opposite of the client authenticating with the server. In this case, the server is saying, "I am server_xyz and here is the certificate that proves it." The client can then utilize that certificate to verify the server is "server_xyz."

There are other important aspects to SSL/TLS that involve the supported encryption ciphers available in each protocol as well as the inherent security of the SSL or TLS protocols. SSL has gone through three iterations - SSLv1, SSLv2 and SSLv3 - all of which are now considered insecure. TLS has gone through three iterations - TLSv1, TLSv1.1 and TLSv1.2 - of which only TLSv1.1 and TLSv1.2 are considered "secure." Ideally, a client should utilize the highest available TLS version it can and the server should support only TLSv1.1 and TLSv1.2. However, most servers must keep TLSv1 for outdated clients.

Almost all modern browsers support both TLSv1.1 and TLSv1.2. However, a client that utilizes HTTPS may not be a browser. The client may be a java application or a python script that communicates with a web server and must negotiate HTTPS/TLS. In this type of a situation, the questions of what is supported and where becomes much more important.

Determining the Supported SSL Ciphers Using the CLI

Before you begin

This section describes how to use the CLI to determine which SSL ciphers are supported.

Step 1 Get the supported ciphers in your openssl environment, shown as follows:

Example:

```
openssl ciphers 'ALL:eNULL'
```

Step 2 Separate the ciphers using sed or some other tool, shown as follows:

Example:

```
openssl ciphers 'ALL:eNULL' | sed -e 's:/\n/g'
```

Step 3 Loop over the ciphers and poll the APIC to see which ones are supported, shown as follows:

Example:

```
openssl s_client -cipher ?<some cipher to test>? -connect <apic ipaddress>:<ssl port, usually 443>
```

See the following example cipher:

Example:

```
openssl s_client -cipher ?ECDHE-ECDSA-AES128-GCM-SHA256? -connect 10.1.1.14:443
```

Note If the response contains `CONNECTED`, then the cipher is supported.



CHAPTER 20

Removing Unwanted `_ui_` Objects



Caution

Changes made through the APIC Basic GUI can be seen, but cannot be modified in the Advanced GUI, and changes made in the Advanced GUI cannot be rendered in the Basic GUI. The Basic GUI is kept synchronized with the NX-OS style CLI, so that if you make a change from the NX-OS style CLI, these changes are rendered in the Basic GUI, and changes made in the Basic GUI are rendered in the NX-OS style CLI, but the same synchronization does not occur between the Advanced GUI and the NX-OS style CLI. See the following examples:

- Do not mix Basic and Advanced GUI modes. If you apply an interface policy to two ports using Advanced mode and then change the settings of one port using Basic mode, your changes might be applied to both ports.
- Do not mix the Advanced GUI and the CLI, when doing per-interface configuration on APIC. Configurations performed in the GUI, may only partially work in the NX-OS CLI.

For example, if you configure a switch port in the GUI at **Tenants > *tenant-name* > Application Profiles > *application-profile-name* > Application EPGs > *EPG-name* > Static Ports > Deploy Static EPG on PC, VPC, or Interface**

Then you use the `show running-config` command in the NX-OS style CLI, you receive output such as:

```
leaf 102
interface ethernet 1/15
switchport trunk allowed vlan 201 tenant t1 application ap1 epg ep1
exit
exit
```

If you use these commands to configure a static port in the NX-OS style CLI, the following error occurs:

```
apic1(config)# leaf 102
apic1(config-leaf)# interface ethernet 1/15
apic1(config-leaf-if)# switchport trunk allowed vlan 201 tenant t1 application ap1 epg
ep1
No vlan-domain associated to node 102 interface ethernet1/15 encap vlan-201
```

This occurs because the CLI has validations that are not performed by the APIC GUI. For the commands from the `show running-config` command to function in the NX-OS CLI, a `vlan-domain` must have been previously configured. The order of configuration is not enforced in the GUI.

- Do not make changes with the Basic GUI or the NX-OS CLI before using the Advanced GUI. This may also inadvertently cause objects to be created (with names prepended with `_ui_`) which cannot be changed or deleted in the Advanced GUI.

If you make changes with the Basic GUI or the NX-OS CLI before using the Advanced GUI, this may inadvertently cause objects to be created (with names prepended with `_ui_`) which cannot be changed or deleted in the Advanced GUI.

For the steps to remove such objects, see [Removing Unwanted `_ui_` Objects Using the REST API, on page 164](#).

- [Removing Unwanted `_ui_` Objects Using the REST API, on page 164](#)

Removing Unwanted `_ui_` Objects Using the REST API

If you make changes with the Cisco NX-OS-Style CLI before using the Cisco APIC GUI, and objects appear in the Cisco APIC GUI (with names prepended with `_ui_`), these objects can be removed by performing a REST API request to the API, containing the following:

- The Class name, for example **infraAccPortGrp**
- The Dn attribute, for example **dn="uni/infra/funcprof/accportgrp-__ui_l101_eth1--31"**
- The Status attribute set to **status="deleted"**

Perform the POST to the API with the following steps:

Step 1 Log on to a user account with write access to the object to be removed.

Step 2 Send a POST to the API such as the following example:

```
POST https://192.168.20.123/api/mo/uni.xml
Payload:<infraAccPortGrp dn="uni/infra/funcprof/accportgrp-__ui_l101_eth1--31" status="deleted"/>
```



CHAPTER 21

Troubleshooting Multipod and Multi-Site Issues

This chapter contains the following sections:

- [Troubleshooting Multi-Site and Multi-Pod, on page 165](#)

Troubleshooting Multi-Site and Multi-Pod

This section describes how to troubleshoot Multi-Site and Multi-Pod.

Error:400

If you receive the following error:

```
Error:400 - Invalid Configuration Following Intersite Spines are not configured as Mpod  
Spines: 1202
```

You must enable the fabric external connectivity for all the existing spines and if you are trying to add new spines use the **Setup Multipod** GUI wizard.

There are two ways to resolve this issue.

- Enable all the spines under the external routed network:
 - In the APIC GUI, on the menu bar, click **Tenant > infra**.
 - In the **Navigation** pane, expand **Networking > External Routed Networks**, right-click on the external routed network and choose **Enable Fabric External Connectivity**.
- Add new spines under the external routed network:
 - In the APIC GUI, on the menu bar, click **Fabric**.
 - In the **Navigation** pane, expand **Quick Start > Node or Pod Setup > Setup Multipod** and complete the Multipod setup.



APPENDIX **A**

acidiag Command

To troubleshoot operations on the Cisco APIC, use the **acidiag** command.



Caution This command is not intended for every day operation of ACI. Running all forms of the command can be very disruptive and cause major issues in your network if not used properly. Make sure you understand the full effect on your fabric before running them.

Cluster Commands

```
acidiag
```

```
acidiag avread
```

```
acidiag fnvread
```

```
acidiag fnvreadex
```

Syntax Description

| Option | Function |
|---------------|---|
| avread | Displays APICs within the cluster. The avread output includes: <ul style="list-style-type: none">• Cluster of —Operational cluster size• out of targeted—The desired cluster size• active= —Indicates whether the APIC is reachable• health= —The overall APIC health summary. Displays services with degraded health scores.• chassisID= —The known chassis IDs for a given APIC. <p>Note Peer chassis IDs can be incorrect for APICs not currently in the cluster.</p> |

| Option | Function |
|--|---|
| bootcurr | On the next boot, the APIC system will boot the current APIC image in the Linux partition. This option is not expected to normally be used. |
| bootother | On the next boot, the APIC system will boot the previous APIC image in the Linux partition. This option is not expected to normally be used. |
| bond0test | Disruptive test of the APIC connection to the leaf. This is used for internal Cisco testing purposes only and outside of that could cause issues with the APIC connection to the fabric. |
| fnvread | Displays the address and state of switch nodes registered with the fabric. |
| fnvreadex | Displays additional information for switch nodes registered with the fabric. |
| linkflap | Brings down and back up a specified APIC interface. |
| preservelogs | APIC will archive current logs. During a normal reboot this automatically occurs. This option can be used prior to a hard reboot. |
| run | Two available options are iptables-list and lldptool. The iptables-list is used to display the Linux iptables, which are controlled by the mgmt Tenant contracts. lldptool is used to display lldp information which is sent or received by the APIC. |
| rvread | Summarizes the data layer state. The output shows a summary of the data layer state for each service. The shard view shows replicas in ascending order. |
| acidiag rvread <i>service</i> | Displays the data layer state for a service on all shards across all replicas.

Note For an example, see Examples, on page 172 |
| acidiag rvread <i>service shard</i> | Displays the data layer state for a service on a specific shard across all replicas.

Note For an example, see Examples, on page 172 |
| acidiag rvread <i>service shard replica</i> | Displays the data layer state for a service on a specific shard and replica.

Note For an example, see Examples, on page 172 |

| Option | Function |
|---------------------------|--|
| validateimage | Prior to loading an image into the firmware repository, the image can be validated. Note that this function runs as a normal part of the process of the image being added into the repository. |
| validateenginxconf | Validates the generated nginx configuration file on APIC to ensure nginx can start with that configuration file. This is meant for debug use, in cases where the nginx webserver is not running on APIC. |

Service IDs

The service IDs listed in the table below are also visible when entering the **man acidiag** command.

Table 3: Service IDs

| Service | ID |
|-------------------|----|
| cliID | 1 |
| controller | 2 |
| eventmgr | 3 |
| extXMLApi | 4 |
| polycyelem | 5 |
| polycymgr | 6 |
| reader | 7 |
| ae | 8 |
| topomgr | 9 |
| observer | 10 |
| dbgr | 11 |
| observerelem | 12 |
| dbgrelem | 13 |
| vmmmgr | 14 |
| nxosmock | 15 |
| bootmgr | 16 |
| appliancedirector | 17 |
| adrelay | 18 |

| Service | ID |
|---------------|----|
| ospaagent | 19 |
| vleafelem | 20 |
| dhcpd | 21 |
| scripthandler | 22 |
| idmgr | 23 |
| ospaelem | 24 |
| osh | 25 |
| opflexagent | 26 |
| opflexelem | 27 |
| confelem | 28 |
| vtap | 29 |
| snmpd | 30 |
| opflexp | 31 |
| analytics | 32 |
| policydist | 33 |
| plghandler | 34 |
| domainmgr | 35 |
| licensemgr | 36 |
| N/A | 37 |
| platformmgr | 38 |
| edmgr | 39 |

Table 4: Data States

| State | ID |
|--------------------------------|----|
| COMATOSE | 0 |
| NEWLY_BORN | 1 |
| UNKNOWN | 2 |
| DATA_LAYER_DIVERGED | 11 |
| DATA_LAYER_DEGRADED_LEADERSHIP | 12 |

| State | ID |
|--|-----|
| DATA_LAYER_ENTIRELY_DIVERGED | 111 |
| DATA_LAYER_PARTIALLY_DIVERGED | 112 |
| DATA_LAYER_ENTIRELY_DEGRADED_LEADERSHIP | 121 |
| DATA_LAYER_PARTIALLY_DEGRADED_LEADERSHIP | 122 |
| FULLY_FIT | 255 |

System Keywords

```
acidiag [{ start | stop | restart }] [{ mgmt | xinetd }]
```

```
acidiag installer -u imageurl -c
```

```
acidiag reboot
```

```
acidiag touch [{ clean | setup }]
```

```
acidiag verifyapic
```

Syntax Description

| Option | Function |
|------------------------------|--|
| -c | Specifies a clean install |
| -u | Specifies a URL for the APIC image. |
| <i>imageurl</i> | Specifies an APIC image. |
| installer | Installs a new image on the APIC, -c for clean install |
| mgmt | Specifies all services on the APIC. |
| reboot | Reboots the APIC. |
| restart | Restarts services on an APIC. |
| start | Starts services on an APIC. |
| stop | Stops services on an APIC. |
| touch [clean setup] | Resets the APIC configuration. <ul style="list-style-type: none"> • The clean option removes all policy data while retaining the APIC network configuration (such as fabric name, IP address, login) • The setup option removes both policy data and the APIC network configuration. |
| verifyapic | Displays the APIC software version. |
| xinetd | Specifies xinetd (extended internet daemon) service, which controls the ssh and telnet daemons. |

Diagnostic Keywords

```
acidiag crashsuspecttracker
```

```
acidiag dbgtoken
```

```
acidiag version
```

| Syntax Description | Option | Function |
|--------------------|----------------------------|--|
| | crashsuspecttracker | Tracks states of a service or data subset that indicate a crash. |
| | dbgtoken | Generates a token used to generate a root password. This is to be used as directed while working with the TAC as needed. |
| | version | Displays the APIC ISO software version. |

Examples

The following examples show how to use the **acidiag** command:

```
apic1# acidiag version 2.2.1o
```

```
apic1# acidiag verifyapic
openssl_check: certificate details
subject= CN=ABC12345678,serialNumber=PID:APIC-SERVER-L1 SN:ABC12345678
issuer= CN=Cisco Manufacturing CA,O=Cisco Systems
notBefore=Sep 28 17:17:42 2016 GMT
notAfter=Sep 28 17:27:42 2026 GMT
openssl_check: passed
ssh_check: passed
all_checks: passed
```

```
apic1# acidiag avread
Local appliance ID=1 ADDRESS=10.0.0.1 TEP ADDRESS=10.0.0.0/16 ROUTABLE IP ADDRESS=0.0.0.0
CHASSIS_ID=1009f750-adab-11e9-a044-8dbd212cd556
Cluster of 7 lm(t):1(2019-08-08T01:02:17.961-07:00) appliances (out of targeted 7
lm(t):7(2019-08-08T03:50:57.240-07:00)) with FABRIC_DOMAIN name=ACI Fabric1 set to
version=apic-4.2(0.235j) lm(t):1(2019-08-17T01:09:16.413-07:00); discoveryMode=PERMISSIVE
lm(t):0(1969-12-31T17:00:00.007-07:00); drrMode=OFF lm(t):0(1969-12-31T17:00:00.007-07:00);
kafkaMode=OFF lm(t):0(1969-12-31T17:00:00.007-07:00)
  appliance id=1 address=10.0.0.1 lm(t):1(2019-08-08T01:02:08.544-07:00) tep
address=10.0.0.0/16 lm(t):1(2019-08-08T01:02:08.544-07:00) routable address=0.0.0.0
lm(t):1(zeroTime) oob address=172.23.96.10/21 lm(t):1(2019-08-08T01:02:18.218-07:00)
version=4.2(0.235j) lm(t):1(2019-08-15T15:22:00.158-07:00)
chassisId=1009f750-adab-11e9-a044-8dbd212cd556 lm(t):1(2019-08-15T15:22:00.158-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X7F lm(t):1(2019-08-17T01:13:46.997-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-08T01:02:18.228-07:00)
cntrlSbst=(APPROVED, FCH1748V0SZ) lm(t):1(2019-08-15T15:22:00.158-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):1(2019-08-08T01:02:08.544-07:00) commissioned=YES lm(t):1(zeroTime) registered=YES
lm(t):1(2019-08-08T01:02:08.544-07:00) standby=NO lm(t):1(2019-08-08T01:02:08.544-07:00)
DRR=NO lm(t):0(zeroTime) apicX=NO lm(t):1(2019-08-08T01:02:08.544-07:00) virtual=NO
```

```

lm(t):1(2019-08-08T01:02:08.544-07:00) active=YES(2019-08-08T01:02:08.544-07:00)
health=(applnc:255 lm(t):1(2019-08-17T01:39:26.296-07:00) svc's)
  appliance id=2 address=10.0.0.2 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):2(2019-07-23T17:51:38.997-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.96.11/21 lm(t):1(2019-08-18T23:14:28.720-07:00)
version=4.2(0.235j) lm(t):2(2019-08-15T15:22:00.300-07:00)
chassisId=694e6a98-adac-11e9-ad79-d1f60e3ee822 lm(t):2(2019-08-15T15:22:00.300-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X2 lm(t):2(2019-08-14T07:55:10.074-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.829-07:00)
cntrlSbst=(APPROVED, FCH1748V0MS) lm(t):2(2019-08-15T15:22:00.300-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):2(2019-08-08T01:42:03.670-07:00) commissioned=YES lm(t):1(2019-08-08T01:02:17.961-07:00)
  registered=YES lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO
lm(t):2(2019-08-08T01:42:03.670-07:00) DRR=NO lm(t):1(2019-08-08T01:02:17.961-07:00) apicX=NO
  lm(t):2(2019-08-08T01:42:03.670-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:02:32.983-07:00) health=(applnc:255
lm(t):2(2019-08-17T01:32:51.454-07:00) svc's)
  appliance id=3 address=10.0.0.3 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):3(2019-07-23T19:05:56.405-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.96.12/21 lm(t):1(2019-08-18T23:14:28.721-07:00)
version=4.2(0.235j) lm(t):3(2019-08-15T15:21:59.893-07:00)
chassisId=1f98b916-adb7-11e9-a6f8-abe00a04e8e6 lm(t):3(2019-08-15T15:21:59.893-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X4 lm(t):3(2019-08-14T07:55:22.256-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
cntrlSbst=(APPROVED, FCH1930V1X6) lm(t):3(2019-08-15T15:21:59.893-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):3(2019-08-08T02:15:20.560-07:00) commissioned=YES lm(t):2(2019-08-08T01:42:15.337-07:00)
  registered=YES lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO
lm(t):3(2019-08-08T02:15:20.560-07:00) DRR=NO lm(t):2(2019-08-08T01:42:15.337-07:00) apicX=NO
  lm(t):3(2019-08-08T02:15:20.560-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:02:33.182-07:00) health=(applnc:255
lm(t):3(2019-08-15T16:08:46.119-07:00) svc's)
  appliance id=4 address=10.0.0.4 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):4(2019-07-23T17:46:15.545-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.97.231/21 lm(t):1(2019-08-18T23:14:28.717-07:00)
version=4.2(0.235j) lm(t):4(2019-08-15T15:22:00.669-07:00)
chassisId=3a7f38aa-adac-11e9-8869-a9e520cdc042 lm(t):4(2019-08-15T15:22:00.669-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X8 lm(t):4(2019-08-14T07:54:59.490-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.825-07:00)
cntrlSbst=(APPROVED, FCH1902V1WW) lm(t):4(2019-08-15T15:22:00.669-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):4(2019-08-08T02:40:09.610-07:00) commissioned=YES lm(t):3(2019-08-08T02:15:32.613-07:00)
  registered=YES lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO
lm(t):4(2019-08-08T02:40:09.610-07:00) DRR=NO lm(t):3(2019-08-08T02:15:32.613-07:00) apicX=NO
  lm(t):4(2019-08-08T02:40:09.610-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-15T15:21:59.914-07:00) health=(applnc:255
lm(t):4(2019-08-17T01:39:26.477-07:00) svc's)
  appliance id=5 address=10.0.0.5 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):5(2019-07-23T19:05:11.089-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.23.97.232/21 lm(t):1(2019-08-18T23:14:28.723-07:00)
version=4.2(0.235j) lm(t):5(2019-08-15T15:22:00.248-07:00)
chassisId=35428666-adb7-11e9-a315-1d7671b518b3 lm(t):5(2019-08-15T15:22:00.248-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X10 lm(t):5(2019-08-14T07:55:19.573-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)

```

```

oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.854-07:00)
cntrlSbst=(APPROVED,FCH1902V1EG) lm(t):5(2019-08-15T15:22:00.248-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=1
lm(t):5(2019-08-08T03:03:50.338-07:00) commissioned=YES lm(t):4(2019-08-08T02:40:15.939-07:00)
  registered=YES lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO
lm(t):5(2019-08-08T03:03:50.338-07:00) DRR=NO lm(t):4(2019-08-08T02:40:15.939-07:00) apicX=NO
  lm(t):5(2019-08-08T03:03:50.338-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-15T15:21:59.756-07:00) health=(applnc:255
lm(t):5(2019-08-17T01:32:43.730-07:00) svc's)
  appliance id=6 address=10.0.0.6 lm(t):7(2019-08-08T03:50:55.470-07:00) tep
address=10.0.0.0/16 lm(t):6(2019-07-23T19:39:41.972-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.31.170.230/21 lm(t):1(2019-08-18T23:14:28.727-07:00)
version=4.2(0.235j) lm(t):6(2019-08-15T15:22:00.562-07:00)
chassisId=066c943a-adbc-11e9-bbed-257398025731 lm(t):6(2019-08-15T15:22:00.562-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X20 lm(t):6(2019-08-14T07:55:20.053-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.820-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.821-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.821-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.821-07:00)
cntrlSbst=(APPROVED,WZP22350JFT) lm(t):6(2019-08-15T15:22:00.562-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=9
lm(t):6(2019-08-08T03:28:11.246-07:00) commissioned=YES lm(t):5(2019-08-08T03:03:57.387-07:00)
  registered=YES lm(t):7(2019-07-24T15:24:25.693-07:00) standby=NO
lm(t):6(2019-08-08T03:28:11.246-07:00) DRR=NO lm(t):5(2019-08-08T03:03:57.387-07:00) apicX=NO
  lm(t):6(2019-08-08T03:28:11.246-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:30:37.663-07:00) health=(applnc:255
lm(t):6(2019-08-15T15:57:05.128-07:00) svc's)
  appliance id=7 address=10.0.0.7 lm(t):7(2019-08-08T03:50:48.149-07:00) tep
address=10.0.0.0/16 lm(t):7(2019-07-24T15:24:19.988-07:00) routable address=0.0.0.0
lm(t):0(zeroTime) oob address=172.31.172.157/21 lm(t):1(2019-08-18T23:14:28.722-07:00)
version=4.2(0.235j) lm(t):7(2019-08-15T15:22:00.539-07:00)
chassisId=859be4ae-ae61-11e9-9840-7d9d67698989 lm(t):7(2019-08-15T15:22:00.539-07:00)
capabilities=0X3EEFFFFFFFFF--0X2020--0X40 lm(t):7(2019-08-14T07:55:23.872-07:00)
rK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
aK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobrK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
oobaK=(stable,present,0X206173722D687373) lm(t):1(2019-08-15T15:22:00.824-07:00)
cntrlSbst=(APPROVED,FCH2051V116) lm(t):7(2019-08-15T15:22:00.539-07:00) (targetMbSn=
lm(t):0(zeroTime), failoverStatus=0 lm(t):0(zeroTime)) podId=10
lm(t):7(2019-08-08T03:50:48.149-07:00) commissioned=YES lm(t):6(2019-08-08T03:28:16.727-07:00)
  registered=YES lm(t):6(2019-07-24T15:27:25.518-07:00) standby=NO
lm(t):7(2019-08-08T03:50:48.149-07:00) DRR=NO lm(t):6(2019-08-08T03:28:16.727-07:00) apicX=NO
  lm(t):7(2019-08-08T03:50:48.149-07:00) virtual=NO lm(t):0(zeroTime)
active=YES(2019-08-13T17:30:45.488-07:00) health=(applnc:255
lm(t):7(2019-08-17T01:39:26.549-07:00) svc's)
-----
clusterTime=<diff=2817 common=2019-08-19T15:33:55.929-07:00
local=2019-08-19T15:33:53.112-07:00 pF=<displForm=0 offsSt=0 offsVlu=-25200
lm(t):7(2019-08-08T03:50:55.925-07:00)>>
-----

```

```

apicl# acidiag rvread 6 3 1
(6,3,1) st:6 lm(t):3(2014-10-16T08:48:20.238+00:00) le: reSt:LEADER voGr:0 cuTerm:0x19
lCoTe:0x18
  lCoIn:0x1800000000001b2a veFiSt:0x31 veFiEn:0x31 lm(t):3(2014-10-16T08:48:20.120+00:00)

  lastUpdt 2014-10-16T09:07:00.214+00:00
-----
clusterTime=<diff=65247252 common=2014-10-16T09:07:01.837+00:00
local=2014-10-15T14:59:34.585+00:00
pF=<displForm=0 offsSt=0 offsVlu=0 lm(t):3(2014-10-16T04:50:08.714+00:00)>>

```

```
apic1# acidiag rvread 6 3
(6,3,1) st:6 lm(t):3(2014-10-16T08:48:20.238+00:00) le: reSt:LEADER voGr:0 cuTerm:0x19
lCoTe:0x18
  lCoIn:0x18000000000001b2a veFiSt:0x31 veFiEn:0x31 lm(t):3(2014-10-16T08:48:20.120+00:00)
  lastUpdt 2014-10-16T09:08:30.240+00:00
(6,3,2) st:6 lm(t):1(2014-10-16T08:47:25.323+00:00) le: reSt:FOLLOWER voGr:0 cuTerm:0x19
lCoTe:0x18
  lCoIn:0x18000000000001b2a veFiSt:0x49 veFiEn:0x49 lm(t):1(2014-10-16T08:48:20.384+00:00)
  lp: clSt:2
    lm(t):1(2014-10-16T08:47:03.286+00:00) dbSt:2 lm(t):1(2014-10-16T08:47:02.143+00:00)
  stMnt:1
    lm(t):0(zeroTime) dbCrTs:2014-10-16T08:47:02.143+00:00 lastUpdt
2014-10-16T08:48:20.384+00:00
(6,3,3) st:6 lm(t):2(2014-10-16T08:47:13.576+00:00) le: reSt:FOLLOWER voGr:0 cuTerm:0x19
lCoTe:0x18
  lCoIn:0x18000000000001b2a veFiSt:0x43 veFiEn:0x43 lm(t):2(2014-10-16T08:48:20.376+00:00)
  lastUpdt 2014-10-16T09:08:30.240+00:00
-----
clusterTime=<diff=65247251 common=2014-10-16T09:08:30.445+00:00
local=2014-10-15T15:01:03.194+00:00
pF=<displForm=0 offsSt=0 offsVlu=0 lm(t):3(2014-10-16T04:50:08.714+00:00)>>
```




APPENDIX **B**

Configuring Export Policies for Troubleshooting

Export policies enable you to export statistics, technical support collections, faults, and events to process core files and debug data from the fabric (the APIC as well as the switch) to any external host.

- [About Exporting Files, on page 177](#)
- [File Export Guidelines and Restrictions, on page 177](#)
- [Configuring a Remote Location, on page 178](#)
- [Sending an On-Demand Tech Support File, on page 180](#)

About Exporting Files

An administrator can configure export policies in the APIC to export statistics, technical support collections, faults and events, to process core files and debug data from the fabric (the APIC as well as the switch) to any external host. The exports can be in a variety of formats, including XML, JSON, web sockets, secure copy protocol (SCP), or HTTP. You can subscribe to exports in streaming, periodic, or on-demand formats.

An administrator can configure policy details such as the transfer protocol, compression algorithm, and frequency of transfer. Policies can be configured by users who are authenticated using AAA. A security mechanism for the actual transfer is based on a username and password. Internally, a policy element handles the triggering of data.

File Export Guidelines and Restrictions

- HTTP export and the streaming API format is supported only with statistics information. Core and tech support data are not supported.
- The destination IP address for exported files cannot be an IPv6 address.
- Do not trigger tech support from more than five nodes simultaneously, especially if they are to be exported into the Cisco Application Policy Infrastructure Controller (APIC) or to an external server with insufficient bandwidth and compute resources.
- To collect tech support from all of the nodes in the fabric periodically, you must create multiple policies. Each policy must cover a subset of the nodes and should be scheduled to trigger in a staggered way (at least 30 minutes apart).

- Do not schedule more than one tech support policy for the same node on the Cisco APIC. Running multiple instances of tech support policies on the same node at the same time can result in a huge consumption of Cisco APIC or switch CPU cycles and the other resources.
- We recommend that you use the regular tech support policy for the nodes placed in maintenance mode instead of the on-demand tech support policy.
- The status of an on-going tech support for the nodes in maintenance mode will not be available in the Cisco APIC GUI in the **Admin > Tech Support > *policy_name* > Operational > Status** section. Based on your selection of **Export to Controller** or **Export Destination** in the tech support policy, you can verify the controller (`/data/techsupport`) or the destination server to confirm that the tech support is being captured.
- Tech support collection from the Cisco APIC can time out when the cores on a leaf switch are busy. The cores can become busy if routing processes such as BGP and platform processes such as HAL hog the CPU. If the tech support collection times out, check for the CPU utilization to see if there is a CPU hog. If there is, you can collect the tech support on the leaf switch directly to avoid the timeout issues.

Configuring a Remote Location

Configuring a Remote Location Using the GUI

This procedure explains how to create a remote location using the APIC GUI.

-
- Step 1** On the menu bar, choose **ADMIN > Import/Export**.
- Step 2** In the navigation pane, right-click **Remote Locations** and choose **Create Remote Location**. The **Create Remote Location** dialog appears.
- Step 3** Enter the appropriate values in the **Create Remote Location** dialog fields.
- Note** For an explanation of a field, click the 'i' icon to display the help file.
- Step 4** When finished entering values in the **Create Remote Location** dialog fields, click **Submit**. You have now created a remote location for backing up your data.
-

Configuring a Remote Location Using the REST API

This procedure explains how to create a remote location using the REST API.

```
<fileRemotePath name="local" host="host or ip" protocol="ftp|scp|sftp" remotePath="path to folder" userName="uname" userPasswd="pwd" />
```

Configuring a Remote Location Using the NX-OS Style CLI

In the ACI fabric, you can configure one or more remote destinations for exporting techsupport or configuration files.

SUMMARY STEPS

1. **configure**
2. **[no] remote path** *remote-path-name*
3. **user** *username*
4. **path** {ftp | scp | sftp} *host* [:*port*] [**remote-directory**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	configure Example: apic1# configure	Enters global configuration mode.
Step 2	[no] remote path <i>remote-path-name</i> Example: apic1(config)# remote path myFiles	Enters configuration mode for a remote path.
Step 3	user <i>username</i> Example: apic1(config-remote)# user admin5	Sets the user name for logging in to the remote server. You are prompted for a password.
Step 4	path {ftp scp sftp} <i>host</i> [: <i>port</i>] [remote-directory] Example: apic1(config-remote)# path sftp filehost.example.com:21 remote-directory /reports/apic	Sets the path and protocol to the remote server. You are prompted for a password.

Examples

This example shows how to configure a remote path for exporting files.

```
apic1# configure
apic1(config)# remote path myFiles
apic1(config-remote)# user admin5
You must reset the password when modifying the path:
Password:
Retype password:
apic1(config-remote)# path sftp filehost.example.com:21 remote-directory /reports/apic
You must reset the password when modifying the path:
Password:
Retype password:
```

Sending an On-Demand Tech Support File

Sending an On-Demand Tech Support File Using the GUI

-
- Step 1** In the menu bar, click **Admin**.
- Step 2** In the submenu bar, click **Import/Export**.
- Step 3** In the **Navigation** pane, expand **Export Policies**.
- Step 4** Right-click **On-demand Tech Support** and choose **Create On-demand Tech Support**.
The **Create On-demand Tech Support** dialog box appears.
- Step 5** Enter the appropriate values in the fields of the **Create On-demand Tech Support** dialog box.
- Note** For an explanation of a field, click the help icon in the **Create On-demand Tech Support** dialog box. The help file opens to a properties description page.
- Step 6** Click **Submit** to send the tech support file.
- Note** On-demand tech support files can be saved to another APIC to balance storage and CPU requirements. To verify the location, click on the On-demand Tech Support policy in the **Navigation** pane, then click the **OPERATIONAL** tab in the **Work** pane. The controller is displayed in the **EXPORT LOCATION** field.
- Step 7** Right-click the policy name and choose **Collect Tech Support**.
- Step 8** Choose **Yes** to begin collecting tech support information.
-

Sending an On-Demand Tech Support File Using the REST API

- Step 1** Set the remote destination for a technical support file using the REST API, by sending a POST with XML such as the following example:

Example:

```
<fileRemotePath userName="" remotePort="22" remotePath="" protocol="sftp" name="ToSupport"
host="192.168.200.2"
dn="uni/fabric/path-ToSupport" descr="">

<fileRsARemoteHostToEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default"/>

</fileRemotePath>
```

- Step 2** Generate an on-demand technical support file using the REST API by sending a POST with XML such as the following:

Example:

```
<dbgexpTechSupOnD upgradeLogs="no" startTime="unspecified" name="Tech_Support_9-20-16"
exportToController="no" endTime="unspecified" dn="uni/fabric/tsod-Tech_Support_9-20-16" descr=""
compression="gzip" category="forwarding" adminSt="untriggered">
  <dbgexpRsExportDest tDn="uni/fabric/path-ToSupport"/>
```

```
<dbgexpRsTsSrc tDn="topology/pod-1/node-102/sys"/>
<dbgexpRsTsSrc tDn="topology/pod-1/node-103/sys"/>
<dbgexpRsTsSrc tDn="topology/pod-1/node-101/sys"/>
<dbgexpRsData tDn="uni/fabric/tscont"/>
</dbgexpTechSupOnD>
<fabricFuncP>
  <fabricCtrlrPGrp name="default">
    <fabricRsApplTechSupOnDemand tnDbgexpTechSupOnDName=" Tech_Support_9-20-16"/>
  </fabricCtrlrPGrp>
</fabricFuncP>
```



APPENDIX **C**

Finding the Switch Inventory

Knowing your switch model and serial numbers can help TAC support with troubleshooting your fabric. This section explains how to find the switch model and serial numbers using the Cisco APIC GUI, CLI, and REST API.

- [Finding Your Switch Inventory Using the GUI, on page 183](#)
- [Finding Your Switch Inventory Using the NX-OS CLI, on page 183](#)
- [Finding Your Switch Inventory Using the REST API, on page 186](#)

Finding Your Switch Inventory Using the GUI

This section explains how to find your switch model and serial numbers using the Cisco APIC GUI.

Before you begin

You must have access to the Cisco APIC GUI

-
- Step 1** On the menu bar, choose **Fabric > Inventory**.
 - Step 2** In the navigation pane, click a **Pod** icon.
Your switch icons appear in the navigation pane.
 - Step 3** In the navigation pane, click on a switch icon.
A list of tabs appears at the top of the work pane.
 - Step 4** Click the **General** tab.
Your switch information appears in the work pane.
-

Finding Your Switch Inventory Using the NX-OS CLI

This section explains how to find your switch model and serial numbers using the NX-OS CLI.

Find your switch inventory as follows:

Example:

```

switch# show hardware
Cisco Nexus Operating System (NX-OS) Software
TAC support: http://www.cisco.com/tac
Documents: http://www.cisco.com/en/US/products/ps9372/tsd_products_support_series_home.html
Copyright (c) 2002-2014, Cisco Systems, Inc. All rights reserved.
The copyrights to certain works contained in this software are
owned by other third parties and used and distributed under
license. Certain components of this software are licensed under
the GNU General Public License (GPL) version 2.0 or the GNU
Lesser General Public License (LGPL) Version 2.1. A copy of each
such license is available at
http://www.opensource.org/licenses/gpl-2.0.php and
http://www.opensource.org/licenses/lgpl-2.1.php

```

Software

```

BIOS:          version 07.56
kickstart:     version 12.1(1h) [build 12.1(1h)]
system:        version 12.1(1h) [build 12.1(1h)]
PE:            version 2.1(1h)
BIOS compile time:      06/08/2016
kickstart image file is: /bootflash/aci-n9000-dk9.12.1.1h.bin
kickstart compile time: 10/01/2016 20:10:40 [10/01/2016 20:10:40]
system image file is:   /bootflash/auto-s
system compile time:    10/01/2016 20:10:40 [10/01/2016 20:10:40]

```

Hardware

```

cisco N9K-C93180YC-EX ("supervisor")
  Intel(R) Xeon(R) CPU @ 1.80GHz with 16400384 kB of memory.
  Processor Board ID FDO20101H1W

```

```

Device name: ifav41-leaf204
bootflash:   62522368 kB

```

Kernel uptime is 02 day(s), 21 hour(s), 42 minute(s), 31 second(s)

Last reset at 241000 usecs after Sun Oct 02 01:27:25 2016

```

Reason: reset-by-installer
System version: 12.1(1e)
Service: Upgrade

```

plugin

```

Core Plugin, Ethernet Plugin
-----

```

Switch hardware ID information

```

-----
Switch is booted up
Switch type is : Nexus C93180YC-EX Chassis
Model number is N9K-C93180YC-EX
H/W version is 0.2010
Part Number is 73-15298-01
Part Revision is 1
Manufacture Date is Year 20 Week 10
Serial number is FDO20101H1W
CLEI code is 73-15298-01

```

```

-----
Chassis has one slot
-----

```

Module1 ok

```

Module type is : 48x10/25G
1 submodules are present

```

```

Model number is N9K-C93180YC-EX
H/W version is 0.2110
Part Number is 73-17776-02
Part Revision is 11
Manufacture Date is Year 20 Week 10
Serial number is FDO20101H1W
CLEI code is 73-17776-02

```

GEM ok

```

Module type is : 6x40/100G Switch
1 submodules are present
Model number is N9K-C93180YC-EX
H/W version is 0.2110
Part Number is 73-17776-02
Part Revision is 11
Manufacture Date is Year 20 Week 10
Serial number is FDO20101H1W
CLEI code is 73-17776-02

```

```

-----
Chassis has 2 PowerSupply Slots
-----

```

PS1 shut

```

Power supply type is : 54.000000W 220v AC
Model number is NXA-PAC-650W-PE
H/W version is 0.0
Part Number is 341-0729-01
Part Revision is A0
Manufacture Date is Year 19 Week 50
Serial number is LIT19500ZEK
CLEI code is 341-0729-01

```

PS2 ok

```

Power supply type is : 54.000000W 220v AC
Model number is NXA-PAC-650W-PE
H/W version is 0.0
Part Number is 341-0729-01
Part Revision is A0
Manufacture Date is Year 19 Week 50
Serial number is LIT19500ZEA
CLEI code is 341-0729-01

```

```

-----
Chassis has 4 Fans
-----

```

FT1 ok

```

Fan1(sys_fan1) (fan_model:NXA-FAN-30CFM-F)
is not available

```

is inserted but info

FT2 ok

```

Fan2(sys_fan2) (fan_model:NXA-FAN-30CFM-F)
is not available

```

is inserted but info

FT3 ok

```

Fan3(sys_fan3) (fan_model:NXA-FAN-30CFM-F)
is not available

```

is inserted but info

FT4 ok

```
Fan4(sys_fan4) (fan_model:NXA-FAN-30CFM-F) is inserted but info
is not available
```

```
=====
```

Finding Your Switch Inventory Using the REST API

This section explains how to find your switch model and serial numbers using the REST API

Find your switch inventory as follows:

Example:

```
GET
https://192.0.20.123/api/node/mo/topology/pod-1.json?query-target=children&target-subtree-class=fabricNode
```

The following response is returned:

```
response:
{
  "totalCount":"8",
  "imdata":
  [{
    "fabricNode":{
      "attributes":{
        "adSt":"on",
        "childAction":"",
        "delayedHeartbeat":"no",
        "dn":"topology/pod-1/node-103",
        "fabricSt":"active",
        "id":"103",
        "lcOwn":"local",
        "modTs":"2016-10-08T14:49:35.665+00:00",
        "model":"N9K-C9396PX",
        "monPolDn":"uni/fabric/monfab-default",
        "name":"leaf3",
        "nameAlias":"",
        "role":"leaf",
        "serial":"TEP-1-103",
        "status":"","uid":"0",
        "vendor":"Cisco Systems, Inc",
        "version":""}
    },
    "fabricNode":{
      "attributes":{
        "adSt":"on",
        "childAction":"",
        "delayedHeartbeat":"no",
        "dn":"topology/pod-1/node-105",
        "fabricSt":"active",
        "id":"105",
        "lcOwn":"local",
        "modTs":"2016-10-08T14:47:52.011+00:00",
        "model":"N9K-C9508",
        "monPolDn":"uni/fabric/monfab-default",
```



```
    "name":"spine2",
    "nameAlias":"","
    "role":"spine",
    "serial":"TEP-1-105","status":"","
    "uid":"0",
    "vendor":"Cisco Systems, Inc",
    "version":""
    ...
  [TRUNCATED]
  ...
}
```



APPENDIX D

Cisco APIC SSD Replacement

Use this procedure to replace the Solid-State Drive (SSD) in Cisco APIC.



Note This procedure should only be performed when there is at least one APIC with a healthy SSD in the cluster, that is fully fit. If all the APIC controllers in the cluster have SSDs that have failed, open a case with the Cisco Technical Assistance Center (TAC).

- [Replacing the Solid-State Drive in Cisco APIC, on page 189](#)

Replacing the Solid-State Drive in Cisco APIC

- Step 1** Decommission the APIC.
- On the menu bar, choose **System > Controllers**.
 - In the **Navigation** pane, expand **Controllers > apic_controller_name > Cluster as Seen by Node**. From this point, select an `apic_controller_name` that is not the controller that is being decommissioned.
 - In the **Work** pane, verify that the **Health State** in the **Active Controllers** summary table indicates the cluster is **Fully Fit** before continuing.
 - In the same **Work** pane, click **Actions > Decommission** after selecting the controller to be decommissioned..
 - Click **Yes**.
The decommissioned controller displays **Unregistered** in the **Operational State** column. The controller is then taken out of service and no longer visible in the **Work** pane.
- Step 2** This step is conditional and is required if your Cisco IMC version is earlier than 2.0(9c). The minimum Cisco IMC version required to replace SSD is 2.0(9c). Upgrade to Cisco IMC version 2.0(9c) if required.
- Use the [Host Upgrade Utility \(HUU\)](#) to upgrade to Cisco IMC version 2.0(9c).
 - Download the `.iso` image. See [Cisco Host Upgrade Utility 2.0\(7\) User Guide](#) to upgrade to Cisco IMC version 2.0(9c).
- Step 3** Physically remove the old SSD if any, and add the new SSD.
- Step 4** Create RAID volume using the newly installed SSD.
- Log in to Cisco IMC.
 - Choose **Storage > Physical Drive** . Select the newly added physical drive.
 - Choose **Storage > Controller Drive Info**, and click **Clear Foreign Config**.

- d) Click **OK**.
- e) Choose **Storage Controller Drive Info**, and click **Create Virtual Drive from Unused Physical Drives**.
- f) Select 0 from the **Raid Level** drop-down list.
- g) Click **Create Virtual Drive**.
- h) Select the newly created virtual drive and click **Initialize**.
- i) Select the **Initialize Type** from the drop-down list and click **Fast Initialize**.

Step 5 Check TPM Status in BIOS or CIMC before installing APIC. If TPM is not activated, enable TPM in BIOS.

Step 6 Install the APIC image using the virtual media. In this step, the SSD is partitioned and the APIC software is installed on the HDD.

Note For a fresh install of Cisco APIC OS 4.x or later, see the *Cisco APIC Installation, Upgrade, and Downgrade Guide*.

- a) Obtain the relevant APIC `.iso` image from CCO. **The image version of the APIC must be the same version as the other Controllers in the cluster.**
- b) Mount the APIC `.iso` image using the Cisco IMC vMedia functionality.
- c) Boot or power cycle the controller.
- d) During the boot process press **F6** to select the **Cisco vKVM-Mapped vDVD** as the one-time boot device. You may be required to enter the BIOS password. The default password is 'password'.
- e) Follow the onscreen instructions to install the APIC software.
- f) After the installation is completed, un-map the virtual media mount.

Step 7 Commission the APIC.

- a) Select any other APIC that is part of the cluster. From the menu bar, choose **SYSTEM > Controllers**.
- b) In the **Navigation** pane, expand **Controllers > apic_controller_name > Cluster as Seen by Node**. From this point, select any `apic_controller_name` that is part of the cluster.
- c) From the **Work** pane, click the decommissioned controller that displaying **Unregistered** in the **Operational State** column.
- d) From the **Work** pane, click **Actions > Commission**.
- e) In the **Confirmation** dialog box, click **Yes**.

The commissioned controller displays the Health state as **Fully-fit** and the operational state as **Available**. The controller should now be visible in the **Work** pane.



APPENDIX E

Expected Output Errors

- [Expected Output Errors, on page 191](#)

Expected Output Errors

Cisco Nexus hardware -EX, -FX1-3, and N93xxC can show output errors on internal interface counters and cause a fault (F119936) to be raised in ACI environments. As long as the output error counters under **show interface** remains unchanged, this is an expected behavior.

Also, note that the **show platform internal counters port** output error will increment. However, if checking the same port with **show interface**, the output error rate will not increment.

This section provides an example of the expected output errors.

```
module-1# show platform internal counters port 51
Stats for port 51
(note: forward drops includes sup redirected packets too)
IF          LPort          Input          Output
           LPort          Packets        Bytes         Packets        Bytes
eth-1/51    51    Total          669974        110547179     692398         194500094
           Unicast          112138        30292113      439809         161274739
           Multicast          0              0             251315         33075023
           Flood            261736        32880023      1274           150332
           Total Drops      296100                261736
           Buffer            0                  0
           Error            0                  261736
           <...>

leaf-101# show interface ethernet 1/51
Ethernet1/51 is up
admin state is up, Dedicated Interface
  Hardware: 1000/10000/100000/40000 Ethernet, address: 0000.0000.0000 (bia a023.9f56.48f3)
  MTU 9366 bytes, BW 40000000 Kbit, DLY 1 usec
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, medium is broadcast
  Port mode is routed
  full-duplex, 40 Gb/s, media type is 40G
  FEC (forward-error-correction) : disable-fec
  Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned off
  Rate mode is dedicated
  Switchport monitor is off
  EtherType is 0x8100
```

```
EEE (efficient-ethernet) : n/a
Last link flapped 1d14h
Last clearing of "show interface" counters never
1 interface resets
30 seconds input rate 4912 bits/sec, 3 packets/sec
30 seconds output rate 1944 bits/sec, 2 packets/sec
Load-Interval #2: 5 minute (300 seconds)
  input rate 3360 bps, 2 pps; output rate 10504 bps, 4 pps
L3 in Switched:
  ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
L3 out Switched:
  ucast: 0 pkts, 0 bytes - mcast: 0 pkts, 0 bytes
RX
  352942 unicast packets  317417 multicast packets  0 broadcast packets
  670359 input packets  110608007 bytes
  8643 jumbo packets  0 storm suppression bytes
  0 runts  0 giants  0 CRC  0 no buffer
  0 input error  0 short frame  0 overrun  0 underrun  0 ignored
  0 watchdog  0 bad etype drop  0 bad proto drop  0 if down drop
  0 input with dribble  0 input discard
  0 Rx pause
TX
  417109 unicast packets  275682 multicast packets  0 broadcast packets
  692791 output packets  194559643 bytes
  7173 jumbo packets
0 output error  0 collision  0 deferred  0 late collision
  0 lost carrier  0 no carrier  0 babble  0 output discard
  0 Tx pause
```



Viewing CRC Error Counters

- [Viewing CRC and Stomped CRC Error Counters, on page 193](#)
- [Viewing CRC Errors Using the GUI, on page 193](#)
- [Viewing CRC Errors Using the CLI, on page 194](#)

Viewing CRC and Stomped CRC Error Counters

Beginning in Cisco APIC Release 4.2(3), CRC errors are split into two categories: CRC errors and stomped CRC errors. CRC errors are corrupted frames that were dropped locally and stomped CRC errors are corrupted frames that were cut-through switched. This differentiation can make it easier to identify the actual interface impacted by CRC errors and troubleshoot physical layer issues within the fabric.

This section demonstrates how to view the CRC and stomped CRC errors.

Viewing CRC Errors Using the GUI

This section demonstrates how to view CRC and stomped CRC error counters using the GUI.

SUMMARY STEPS

1. On the menu bar, choose **Fabric > Inventory**.
2. In the **Navigation** pane, click to expand a pod.
3. Click to expand **Interfaces**.
4. Click to choose an interface.
5. In the **Work** pane, click the **Error Counters** tab.

DETAILED STEPS

- Step 1** On the menu bar, choose **Fabric > Inventory**.
- Step 2** In the **Navigation** pane, click to expand a pod.
- Step 3** Click to expand **Interfaces**.
A list of interfaces appear in the **Navigation** pane.
- Step 4** Click to choose an interface.
A list of tabs appear in the **Work** pane.

- Step 5** In the **Work** pane, click the **Error Counters** tab.
A list of error categories appears including **CRC Errors (FCS Errors)** and **Stomped CRC Errors (packets)**.

Viewing CRC Errors Using the CLI

This section demonstrates how to view CRC and stomped CRC error counters using the CLI

To view CRC and stomped CRC errors:

Example:

```
Switch# show interface ethernet 1/1
Ethernet1/1 is up
admin state is up, Dedicated Interface
  Belongs to po4
  Hardware: 100/1000/10000/25000/auto Ethernet, address: 00a6.cab6.bda5 (bia 00a6.cab6.bda5)
  MTU 9000 bytes, BW 10000000 Kbit, DLY 1 usec
  reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, medium is broadcast
  Port mode is trunk
  full-duplex, 10 Gb/s, media type is 10G
  FEC (forward-error-correction) : disable-fec
^[[B Beacon is turned off
  Auto-Negotiation is turned on
  Input flow-control is off, output flow-control is off
  Auto-mdix is turned off
  Rate mode is dedicated
  Switchport monitor is off
  EtherType is 0x8100
  EEE (efficient-ethernet) : n/a
  Last link flapped 3d02h
  Last clearing of "show interface" counters never
  1 interface resets
  30 seconds input rate 0 bits/sec, 0 packets/sec
  30 seconds output rate 4992 bits/sec, 8 packets/sec
  Load-Interval #2: 5 minute (300 seconds)
    input rate 0 bps, 0 pps; output rate 4536 bps, 8 pps
RX
  0 unicast packets 200563 multicast packets 0 broadcast packets
  200563 input packets 27949761 bytes
  0 jumbo packets 0 storm suppression bytes
  0 runts 0 giants 0 CRC 0 Stomped CRC 0 no buffer
  0 input error 0 short frame 0 overrun 0 underrun 0 ignored
  0 watchdog 0 bad etype drop 0 bad proto drop 0 if down drop
  0 input with dribble 0 input discard
  0 input buffer drop 0 input total drop
  0 Rx pause
TX
  0 unicast packets 2156812 multicast packets 0 broadcast packets
  2156812 output packets 151413837 bytes
  0 jumbo packets
  0 output error 0 collision 0 deferred 0 late collision
  0 lost carrier 0 no carrier 0 babble 0 output discard
  0 output buffer drops 0 output total drops
  0 Tx pause
```




INDEX

(enabling) NX-OS Format [84](#)
(enabling) Syslog [84](#)

A

acidiag Command [167](#)
ACL deny logging [27–28](#)
ACL permit and deny logs [29](#)
ACL permit logging [25–26](#)
atomic counters [33–36, 91, 100](#)
 about [91, 100](#)
 configuring [34](#)
 guidelines and restrictions [33](#)

C

Cisco APIC [122](#)
configuration sync [122](#)
configuration synchronization [122](#)
Contract permit logging [25–26](#)
Contract permit logs [29](#)
core files [177](#)

D

digital optical monitoring [36](#)
Digital Optical Monitoring [37–39](#)
DOM [36–39](#)

E

endpoint connectivity [135](#)
eventlog Command [102–114, 116–121](#)
exporting files [177](#)
 about [177](#)

P

permit logging [28, 30](#)

S

SNMP [44–45, 47–48](#)
 about [44](#)
 configuring policy [45](#)
 configuring trap destination [47](#)
 configuring trap source [48](#)
SPAN [49–50, 54](#)
 about [49](#)
 configuring [54](#)
 guidelines and restrictions [50](#)
syslog [81–83](#)
 about [81](#)
 destination [82](#)
 source [83](#)

T

Taboo Contract deny logging [27–28](#)
Taboo Contract deny logs [29](#)
Taboo Contract drop logging [27–28](#)
techsupport file [180](#)
 sending [180](#)
techsupport files [177](#)
traceroute [85, 87–88](#)
 about [85](#)
 configuring [88](#)
 guidelines and restrictions [87](#)
troubleshooting [122](#)

