ACI Bootcamp Lab

Fabric External Data Collector Policies Configuration - SNMP

ACI Solutions Team by Tomas de Leon (tdeleon@cisco.com)

TABLE OF CONTENTS

1 Introduction

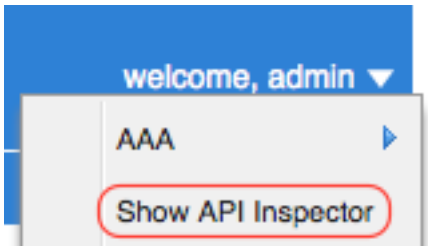
The SNMP protocol governs the network management and monitoring of your network devices. ACI provides extensive SNMPv1, v2, and v3 support, including Management Information Bases (MIBs) and notifications (traps). The SNMP standard allows any third-party applications that support the different MIBs to manage and monitor the ACI fabric.

SNMPv3 provides extended security. Each 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. For more information about using SNMP, see the Cisco ACI MIB Quick Reference.

Note: The SNMP policy is applied to individual switches. However, the SNMP policy source is created as a monitoring policy.

The following Lab involves using the "SNMP" utility to gather information about your Cisco ACI fabric system. The "SNMP" utility will send SNMP traps about your designated ACI Fabric. The Nodes in ACI Fabric sends SNMP Trap messages to SNMP Trap receivers during operation. Not all SNMP Traps 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. This lab will show examples of configuring SNMP utilizing the APIC Admin GUI and REST API (using POSTMAN). *For this Lab, refer to Section 2 for your designated the fabric topology and your Pod's Application Server.*

In regards to the REST API examples listed in this Lab, there is an assumption made that you have a REST CLIENT (like POSTMAN) installed on your workstation. This will be used for executing REST API requests to an APIC Controller. Also, while executing lab, you may want to open the API inspector console from the APIC GUI. The API inspector displays the API POST requests used for the tasks performed. The Post Requests in the API inspector can be used for sending requests to APIC controllers.



For this Lab, each Pod has an application server that is running a SNMP utility application. The iReasoning MIB Browser Server is installed so that the Cisco ACI Fabric system can send SNMP Traps messages to.

iReasoning MIB Browser Free Personal Edition
http://ireasoning.com/download.shtml

Note: This Lab will show you how to config the Cisco APIC for SNMP external data collectors. For more information on this feature, please refer to the Cisco ACI System Configuration & Reference Guides.

2 Lab Reference & Topology Information

For the following sections in this lab, please use the following fabric information for the POD1 in your fabric pod assignments.

Device\Entity	NodeID	Fabric 1	Fabric 2
APIC 1 (OOB IP Address)	1	10.122.254.211	10.122.254.141
APIC 2 (OOB IP Address)	2	10.122.254.212	10.122.254.142
APIC 3 (OOB IP Address)	3	10.122.254.213	10.122.254.143
Spine 1 (OOB IP Address)	201	10.122.254.244	10.122.254.130
Spine 2 (OOB IP Address)	202	10.122.254.245	10.122.254.131
Leaf 1 (OOB IP Address)	101	10.122.254.241	10.122.254.128
Leaf 2 (OOB IP Address)	102	10.122.254.242	10.122.254.135
Leaf 3 (OOB IP Address)	103	10.122.254.243	10.122.254.136
Leaf 4 (OOB IP Address)	104	10.122.254.154	10.122.254.137
OOB Default Gateway		10.122.254.1 / 24	10.122.254.1 / 24

Device\Entity	IP ADDRESS	RDP Username\Password
ACI-P1-Server (SNMP)	10.122.254.77	Administrator\Cisco123!
ACI-P2-Server (SNMP)	10.122.254.78	Administrator\Cisco123!
ACI-P3-Server (SNMP)	10.122.254.79	Administrator\Cisco123!
ACI-P4-Server (SNMP)	10.122.254.120	Administrator\Cisco123!
ACI-P5-Server (SNMP)	10.122.254.207	Administrator\Cisco123!
ACI-P6-Server (SNMP)	10.122.254.208	Administrator\Cisco123!

SNMP AGENTS	Fabric 1	Fabric 2
Spine 1 (OOB IP Address)	10.122.254.244	10.122.254.130
Spine 2 (OOB IP Address)	10.122.254.245	10.122.254.131
Leaf 1 (OOB IP Address)	10.122.254.241	10.122.254.128
Leaf 2 (OOB IP Address)	10.122.254.242	10.122.254.135
Leaf 3 (OOB IP Address)	10.122.254.243	10.122.254.136
Leaf 4 (OOB IP Address)	10.122.254.154	10.122.254.137

Note: For this SNMP, The SNMP COMMUNITY is "bootcamp" and the SNMP CONTEXT is "bootcamp"

3 Configure SNMP Agents on iReasoning MIB Browser

For this lab section, you will configure SNMP Agents on your Pod's iReasoning MIB Browser Server. The Server application is preconfigured. But Once you are assigned an ACI Fabric for the Lab, you will need to configure the SNMP Agents for the Leaf\Spine Switch addresses in your Pod's iReasoning MIB Browser Server Application. This is necessary for the application to receive SNMP Traps from the ACI Fabric and for performing SNMP Walks to your Leaf\Spine Switches.

This lab section will:

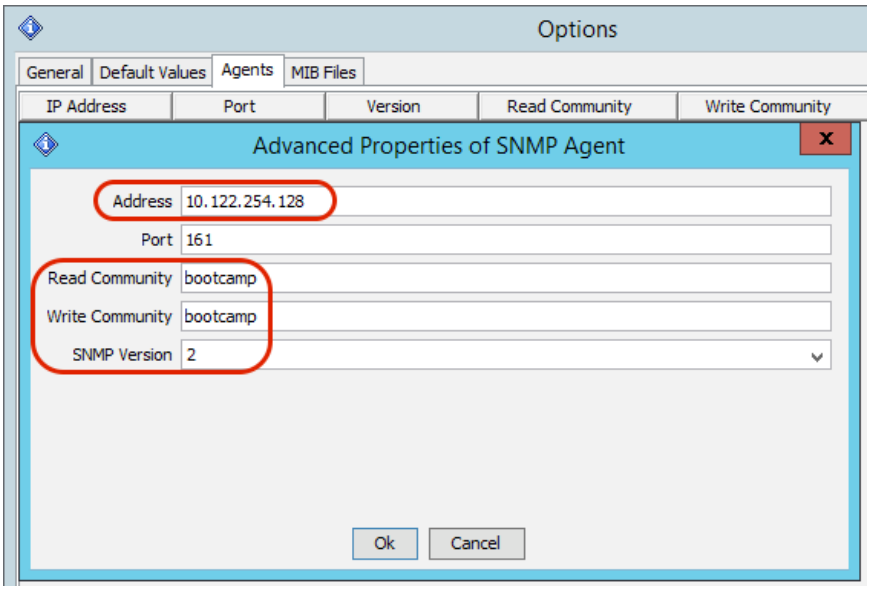
- **Configure SNMP Agents on iReasoning MIB Browser Server.**

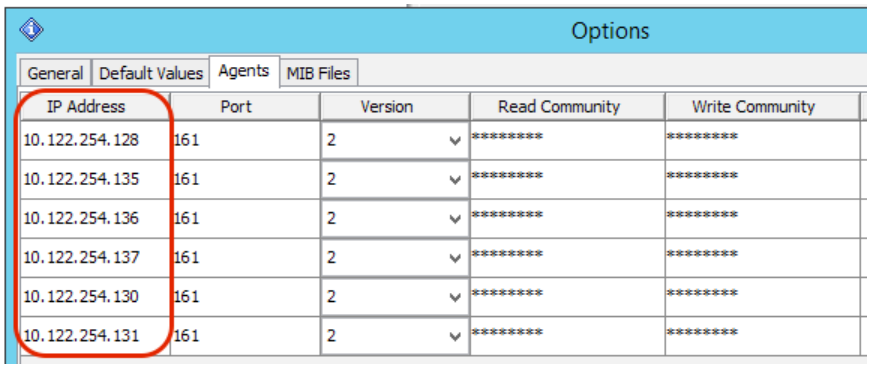
Note: Refer to Section 2 Lab Reference & Topology Information if needed for this Lab.

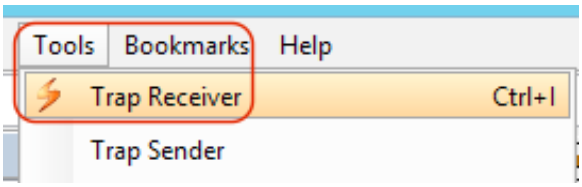
- Remote Desktop to your assigned Pod Application Server
- Hover over the WORLD icon (iReasoning MIB Browser Application) on the TASK BAR. Select the OPEN application console. *If the iReasoning MIB Browser Application is not open, simply open a MIB Browser instance.*
- Configure the SNMP Agents with the Leaf\Spine Switch addresses.

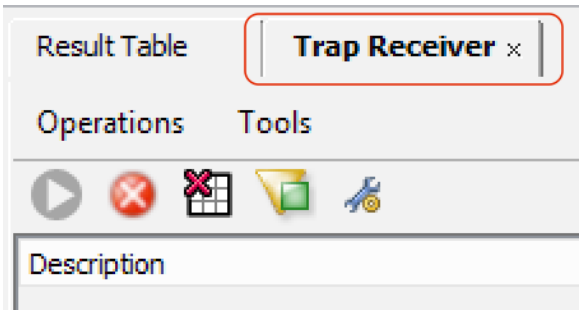
- On the menu bar, choose **TOOLS > OPTIONS**. In **OPTIONS** dialog box, perform the following actions:
  - Select AGENTS Tab
  - Click ADD Button
  - In the ADVANCED PROPERTIES OF SNMP AGENT dialog box, perform the following actions:
    - Enter **Address** (10.122.254.128)
    - Enter **Port** (161)
    - Enter **Read & Write Community** (bootcamp)
    - Select SNMP Version (2)
    - Click **OK**
    - REPEAT Steps above until all Leaf\Spine Switch addresses for your designated Fabric are added as SNMP Agents.
    - Once all Leaf\Spine Switch addresses for your designated Fabric are added as SNMP Agents, Click **OK**

- Enable SNMP Trap Receiver for your iReasoning MIB Browser Application
  - On the menu bar, choose **TOOLS > TRAP RECEIVER.** A **Trap Receiver Tab** will open (if not already) in your MIB Browser Results area of the SNMP Console.




Options
General
Default Values
Agents
MIB Files
IP Address
Port
Version
Read Community
Write Community
10.122.254.128 161 2 ********* *********
10.122.254.135 161 2 ********* *********
10.122.254.136 161 2 ********* *********
10.122.254.137 161 2 ********* *********
10.122.254.130 161 2 ********* *********
10.122.254.131 161 2 ********* *********


Tools
Bookmarks
Help
Trap Receiver
Ctrl+I
Trap Sender


Result Table
Trap Receiver
Operations
Tools
Description

4 Configure SNMP Trap Destinations for the ACI Fabric

For this lab section, you will configure Monitoring Destinations for SNMP Services for Leaf\Spine Switches in your designated ACI fabric.

This lab section will:

- **Create Syslog Monitoring Destination Group.**

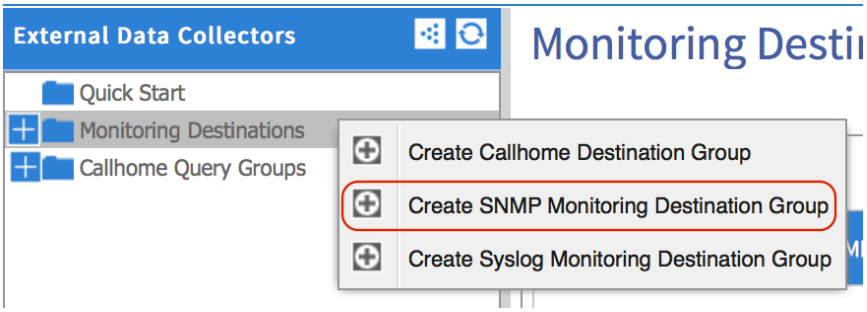
Note: Refer to Section 2 Lab Reference & Topology Information if needed for this Lab.

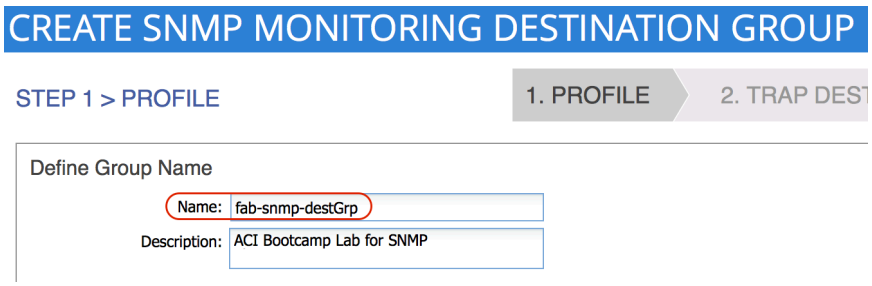
- On the menu bar, choose **ADMIN > EXTERNAL DATA COLLECTORS**. In the Navigation pane, select **MONITORING DESTINATIONS**. Right-click and click **CREATE SNMP MONITORING DESTINATION GROUP**. In the Create SNMP monitoring destination group dialog box, perform the following actions:
    - Enter **Name** (fab-snmp-destGrp)
    - Enter **Description** (ACI Bootcamp Lab for SNMP)
    - Click **NEXT**
    - From the "Create SNMP Monitoring Destination Group" wizard, Create SNMP TRAP Destinations. Click on the " **+** " to **CREATE A SNMP TRAP DESTINATION**. In the Create SNMP TRAP Destination dialog box, perform the following actions:
        - Enter **Host Name/IP** (10.122.254.77 for ACI-P1-Server. Refer to your Application Server address)
        - Enter **Port** (162)
        - Select **Version** (v2c)
        - Enter **Security Name** (bootcamp) *** Community Password*
        - Select **v3 Security Level** (noauth)
        - Select **Management EPG** (default (Out-of-Band))

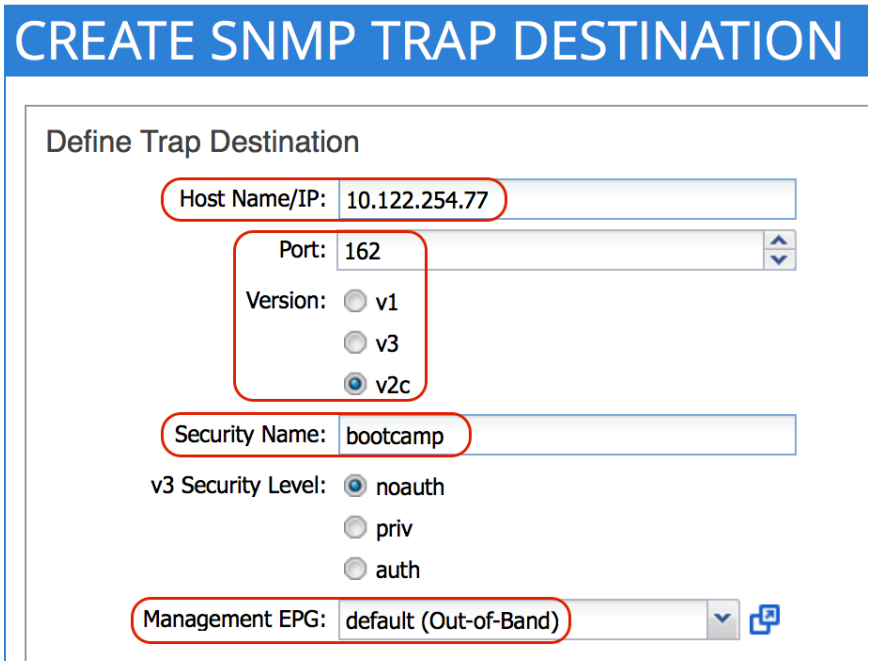
        Note: The Host can be an IP Address or DNS Name. DNS Services for the Fabric must be configured to use DNS Host Names.

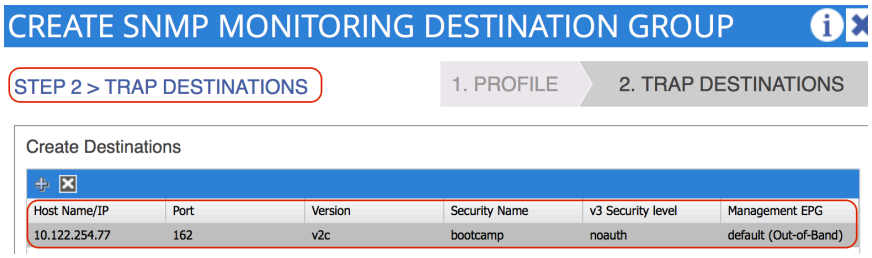
        - Click **OK**

        Note: Since there are three Lab Pods per Fabric, your can add another SNMP TRAP DESTINATION for the OTHER two Pod Servers for your designated Fabric.


External Data Collectors
Monitoring Destin
Quick Start
Monitoring Destinations
Callhome Query Groups
Create Callhome Destination Group
Create SNMP Monitoring Destination Group
Create Syslog Monitoring Destination Group


CREATE SNMP MONITORING DESTINATION GROUP
STEP 1 > PROFILE
1. PROFILE
2. TRAP DEST
Define Group Name
Name: fab-snmp-destGrp
Description: ACI Bootcamp Lab for SNMP


CREATE SNMP TRAP DESTINATION
Define Trap Destination
Host Name/IP: 10.122.254.77
Port: 162
Version: v1
v3
v2c
Security Name: bootcamp
v3 Security Level: noauth
priv
auth
Management EPG: default (Out-of-Band)


CREATE SNMP MONITORING DESTINATION GROUP
STEP 2 > TRAP DESTINATIONS
1. PROFILE
2. TRAP DESTINATIONS
Create Destinations
Host Name/IP
Port
Version
Security Name
v3 Security level
Management EPG
10.122.254.77
162
v2c
bootcamp
noauth
default (Out-of-Band)

Monitoring Destinations

NAME	DESCRIPTION
fab-snmp-destGrp	ACI Bootcamp Lab for SNMP

SNMP Trap Destinations

ACTIONS

HOST	PORT	VERSION	SECURITY NAME	V3 SECURITY LEVEL	MANAGEMENT EPG
10.122.254.77	162	v2c	bootcamp	noauth	default (Out-of-Band)

Using the **APIC API Inspector**, this API Example was captured from the POST request to Create SNMP Monitoring Destination Group. You can use this APIC Example and use POSTMAN REST Client to create the SNMP monitoring destination group.

API EXAMPLE

method: POST

url:

https://10.122.254.141/api/node/mo/uni/fabric/snmpgroup-fab-snmp-destGrp.json

payload

{"snmpGroup":{"attributes":{"dn":"uni/fabric/snmpgroup-fab-snmp-destGrp","name":"fab-snmp-destGrp","descr":"ACI Bootcamp Lab for SNMP","rn":"snmpgroup-fab-snmp-destGrp","status":"created"},"children":[{"snmpTrapDest":{"attributes":{"dn":"uni/fabric/snmpgroup-fab-snmp-destGrp/trapdest-10.122.254.77 -port-162","host":"10.122.254.77","secName":"bootcamp","rn":"trapdest-10.122.254.77 -port-162","status":"created"},"children":[{"fileRsARemoteHostToEpg":{"attributes":{"tDn":"uni/tn-mgmt/mgmtp-default/oob-default","status":"created"},"children":[]}}]}}]}}

5 Configure SNMP Policy for the ACI Fabric

For this lab section, you will configure SNMP Pod Policies for the Leaf\Spine Switches in your designated ACI fabric. Pod policies enable you to configure various functions relating to a pod such as global date and time policies, communication policies, and SNMP.

This lab section will:

- **Create SNMP Policy.**
- **Add SNMP Policy to Pod Policy Group.**

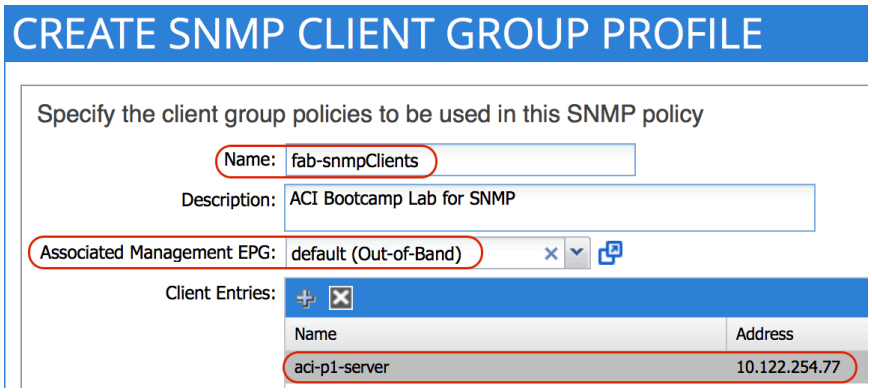
Note: Refer to Section 2 Lab Reference & Topology Information if needed for this Lab.

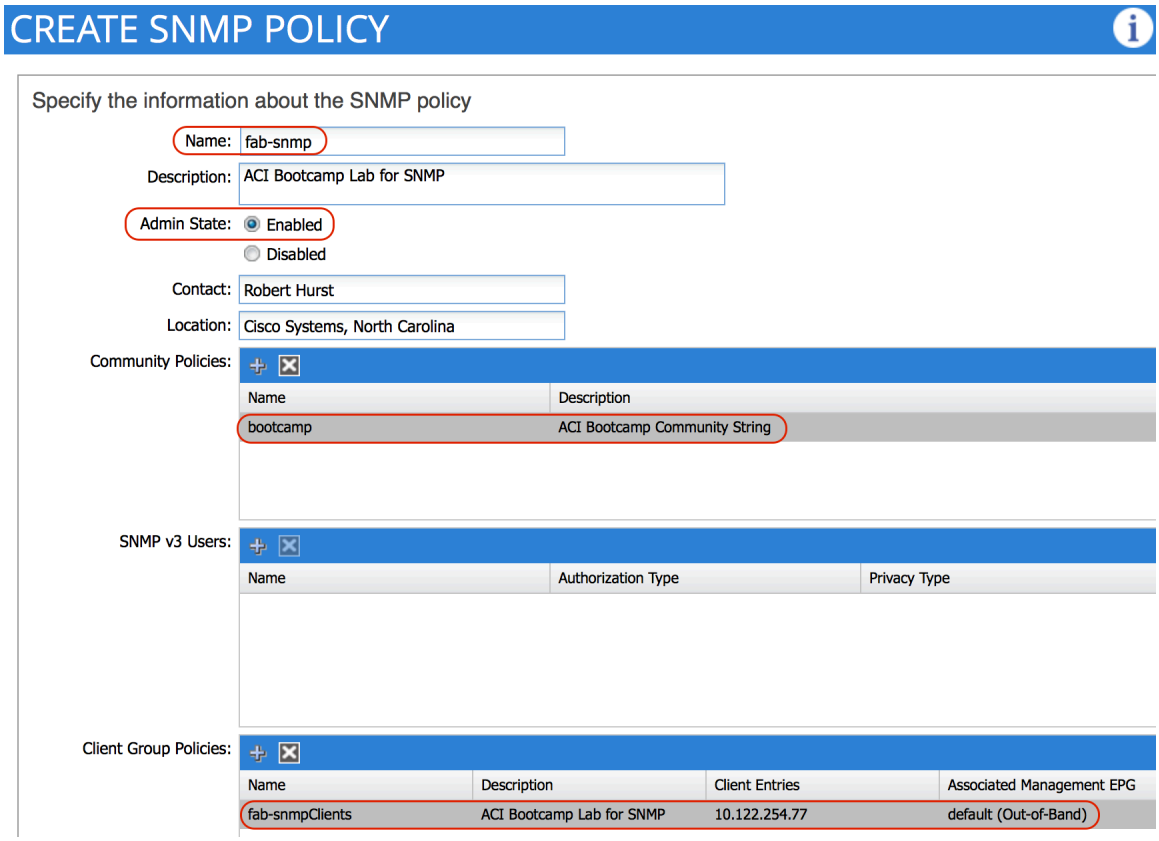
Task 5.1 Use the GUI to a Create SNMP Policy. For this task, use the admin user "admin" and the password "Aci123bc".

- On the menu bar, choose **FABRIC > FABRIC POLICIES**. In the Navigation pane, expand **POD POLICIES**.
- Expand **Policies**
- Select "**SNMP**" and Right Click and Click **CREATE SNMP POLICY**. In the Create SNMP Policy dialog box, perform the following actions:
    - Enter **Name** (fab-snmp)
    - Select **Admin State** (Enabled)
    - Enter **Contact** (Robert Hurst)
    - Enter **Location** (Cisco Systems, North Carolina)
    - Click on the " **+** " sign to **ADD COMMUNITY POLICIES**. In the Community Policies Table, perform the following actions:
        - Enter **Name** (bootcamp)
        - Click **UPDATE**
    - Click on the " **+** " sign to **ADD CLIENT GROUP POLICIES**. In the Client Group Profile dialog box, perform the following actions:
        - Enter **Name** (fab-snmpClients)
        - Select **Management EPG** (default (Out-of-Band))
        - Click on the " **+** " sign to **ADD CLIENT ENTRIES**. In the Client Entries Table, perform the following actions
            - Enter **Name** (aci-p1-server)
            - Enter **Address** (10.122.254.77)
            - Click **UPDATE**

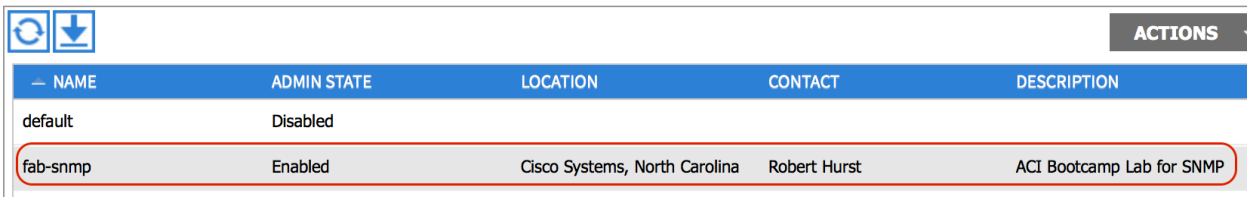
            Note: Since there are three Lab Pods per Fabric, your can add another SNMP CLIENT ENTRIES for the OTHER two Pod Servers for your designated Fabric.

- Click **SUBMIT**





Policies - SNMP



ACTIONS

NAME	ADMIN STATE	LOCATION	CONTACT	DESCRIPTION
default	Disabled			
fab-snmp	Enabled	Cisco Systems, North Carolina	Robert Hurst	ACI Bootcamp Lab for SNMP

Using the **APIC API Inspector**, this API Example was captured from the POST request to Create SNMP Policy. You can use this APIC Example and use POSTMAN REST Client to Create SNMP Policy.

API EXAMPLE

method: POST

url:

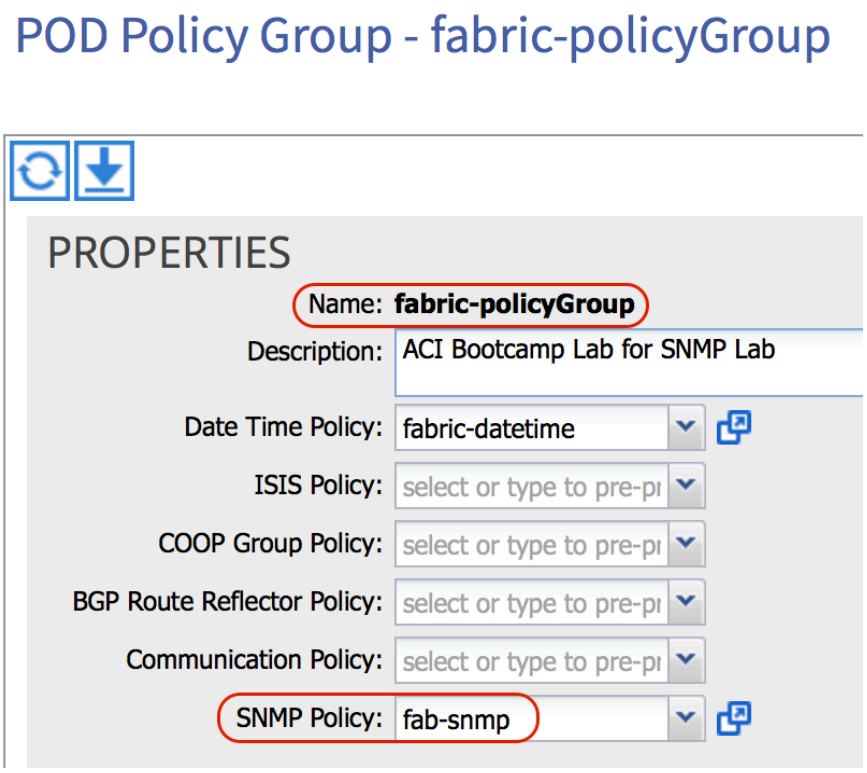
https://10.122.254.141/api/node/mo/uni/fabric/snmppol-fab-snmp.json

payload

{"snmpPol":{"attributes":{"dn":"uni/fabric/snmppol-fab-snmp","name":"fab-snmp","descr":"ACI Bootcamp Lab for SNMP","adminSt":"enabled","contact":"Robert Hurst","loc":"Cisco Systems, North Carolina","rn":"snmppol-fab-snmp","status":"created"},"children":[{"snmpClientGrpP":{"attributes":{"dn":"uni/fabric/snmppol-fab-snmp/clgrp-fab-snmpClients","name":"fab-snmpClients","descr":"ACI Bootcamp Lab for SNMP","rn":"clgrp-fab-snmpClients","status":"created"},"children":[{"snmpClientP":{"attributes":{"dn":"uni/fabric/snmppol-fab-snmp/clgrp-fab-snmpClients/client-[10.122.254.77]","name":"aci-p1-server","addr":"10.122.254.77","rn":"client-[10.122.254.77]","status":"created"},"children":[]}},{"snmpRsEpg":{"attributes":{"tDn":"uni/tn-mgmt/mgmtp-default/oob-default","status":"created"},"children":[]}}]}},{"snmpCommunityP":{"attributes":{"dn":"uni/fabric/snmppol-fab-snmp/community-bootcamp","name":"bootcamp","descr":"ACI Bootcamp Community String","rn":"community-bootcamp","status":"created"},"children":[]}}]}}

Task 5. 2 Use the GUI to Add SNMP Policy to Pod Policy Group. For this task, use the admin user "admin" and the password "Aci123bc".

- On the menu bar, choose **FABRIC > FABRIC POLICIES**. In the Navigation pane, expand **POD POLICIES**.
- Expand **Policies**
- Expand **POLICY GROUPS** and Select **FABRIC-POLICYGROUP**. In the “fabric-policyGroup” Work Pane, Select **SNMP POLICY** (fab-snmp)
- Click **SUBMIT**



Using the **APIC API Inspector**, this API Example was captured from the POST request to Add SNMP Policy to Pod Policy Group. You can use this APIC Example and use POSTMAN REST Client to Add SNMP Policy to Pod Policy Group.

API EXAMPLE

method: POST

url: https://10.122.254.141/api/node/mo/uni/fabric/funcprof/podpgrp-fabric-policyGroup.json

payload

{"fabricPodPGrp":{"attributes":{"dn":"uni/fabric/funcprof/podpgrp-fabric-policyGroup","descr":"ACI Bootcamp Lab for SNMP Lab"},"children":[{"fabricRsSnmpPol":{"attributes":{"tnSnmpPolName":"fab-snmp"},"children":[]}}]}}

6 Configure Monitoring Policies for SNMP in the ACI Fabric

For this lab section, you will configure SNMP Monitoring Policies for Leaf\Spine Switches in your designated ACI fabric.

This lab section will:

- **Configure FABRIC > FABRIC POLICIES to send SNMP TRAPS to SNMP TRAP Destinations.**
- **Configure FABRIC > ACCESS POLICIES to send SNMP TRAPS to SNMP TRAP Destinations.**

Note: Refer to Section 2 Lab Reference & Topology Information if needed for this Lab.

Task 6.1: Configure FABRIC > FABRIC POLICIES to send SNMP TRAPS to SNMP Trap Destinations.

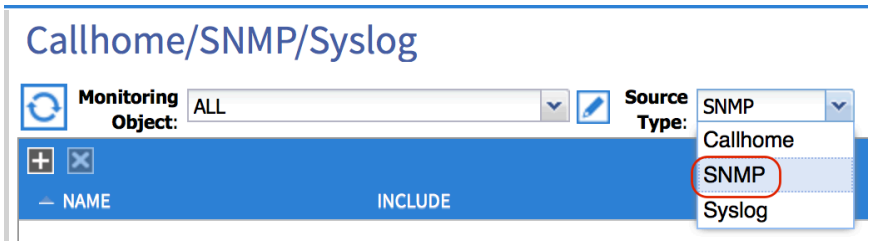
Fabric policies govern the operation of internal fabric interfaces. The system provides default fabric policies. Fabric policies enable configuring various functions or protocols. Administrators who have fabric administrator privileges can create new fabric policies according to their requirements. The APIC enables administrators to select the pods, leaf switches, and interfaces to which they will apply access policies.

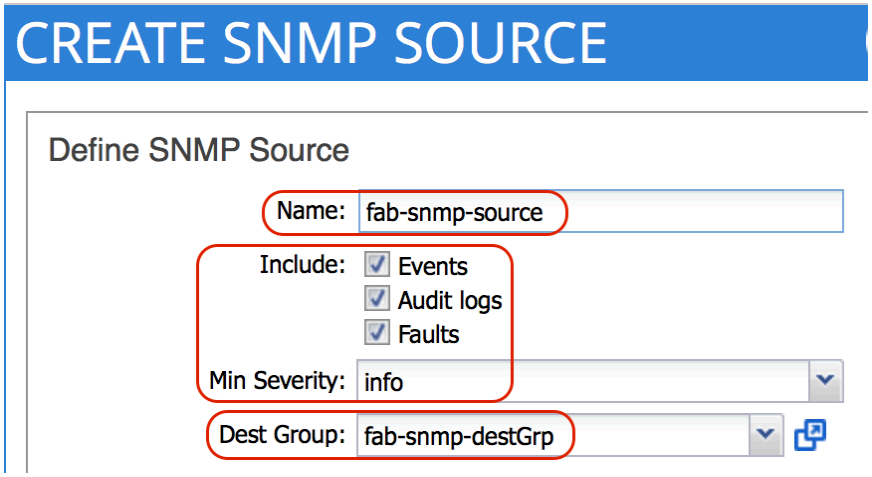
Fabric policies configure interfaces that connect spine and leaf switches. Fabric policies can enable features such as monitoring (statistics collection and statistics export), troubleshooting (on-demand diagnostics and SPAN), or NTP.

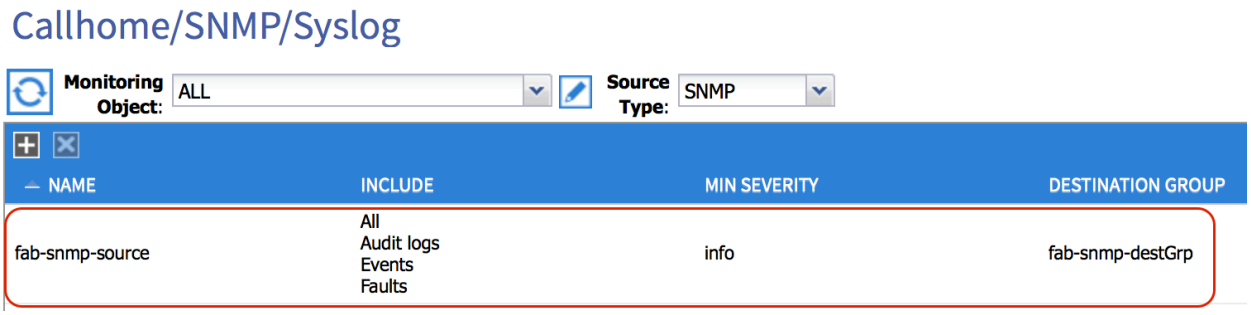
Fabric SNMP Sources need to be configured in the **DEFAULT** and **COMMON** monitoring policies configured in the **Fabric Policies** configuration. Use the GUI to configure the **DEFAULT** and **COMMON** SNMP monitoring policies. Use the API Inspector to capture the API POST information from this configuration.

Task 6.1.1 Use the GUI to a configure the "**DEFAULT**" monitoring policy for SNMP. For this task, use the admin user "admin" and the password "Aci123bc".

- On the menu bar, choose **FABRIC > FABRIC POLICIES**. In the Navigation pane, expand **MONITORING POLICIES**.
- Expand **default**
- Select "**Callhome/SNMP/Syslog**"
- In the "Callhome/SNMP/Syslog" Work Pane, **Select** the **SOURCE TYPE "SNMP"** from the Source Type drop down list.
- Click on the " **+** " sign to **CREATE SNMP SOURCE**. In the Create SNMP Source dialog box, perform the following actions:
    - Enter **Name** (fab-snmp-source)
    - For **Include** (Check boxes for Events, Audit logs, and Faults)
    - Select **Min Severity** (information)
    - Select **Dest Group** (fab-snmp-destGrp)
    - Click **SUBMIT**







Using the **APIC API** Inspector, this API Example was captured from the POST request to create SNMP Source for the "**DEFAULT**" monitoring policy for SNMP. You can use this APIC Example and use POSTMAN REST Client to create SNMP Source for the "DEFAULT" monitoring policy.

<u>API EXAMPLE</u>

method: POST

url:

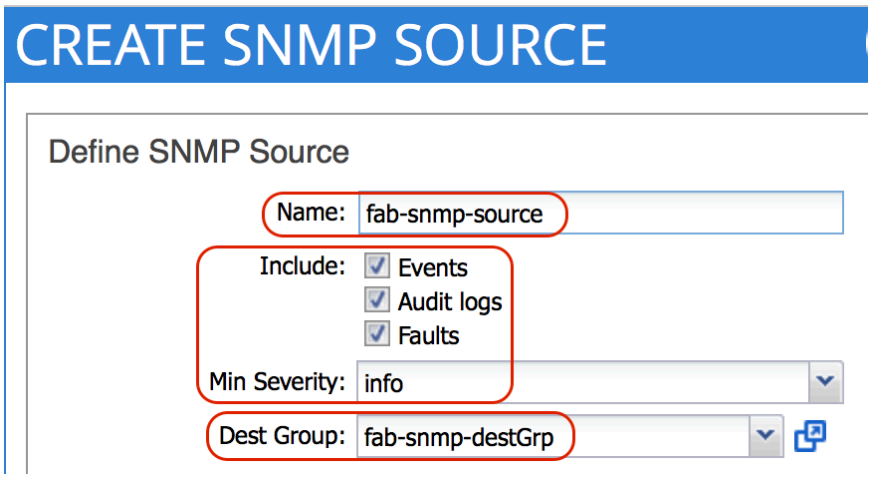
https://10.122.254.141/api/node/mo/uni/fabric/monfab-default/snmpsrc-fab-snmp-source.json

payload

{"snmpSrc":{"attributes":{"dn":"uni/fabric/monfab-default/snmpsrc-fab-snmp-source","name":"fab-snmp-source","incl":"events,audit,faults","rn":"snmpsrc-fab-snmp-source","status":"created"},"children":[{"snmpRsDestGroup":{"attributes":{"tDn":"uni/fabric/snmpgroup-fab-snmp-destGrp","status":"created"},"children":[]}}]}}

Task 6.1.2 Use the GUI to a configure the "**COMMON**" monitoring policy for SNMP. For this task, use the admin user "admin" and the password "Aci123bc".

- On the menu bar, choose **FABRIC > FABRIC POLICIES**. In the Navigation pane, expand **MONITORING POLICIES**.
- Expand **Common Policy**
- Select "**Callhome/SNMP/Syslog**"
- **Right Click** on "Callhome/SNMP/Syslog" and Select the **CREATE SNMP SOURCE**. In the Create SNMP Source dialog box, perform the following actions:
    - Enter **Name** (fab-snmp-source)
    - For **Include** (Check boxes for Events, Audit logs, and Faults)
    - Select **Min Severity** (information)
    - Select **Dest Group** (fab-snmp-destGrp)
    - Click **SUBMIT**



Using the **APIC API** Inspector, this API Example was captured from the POST request to create SNMP Source for the "**COMMON**" monitoring policy for SNMP. You can use this APIC Example and use POSTMAN REST Client to create SNMP Source for the "COMMON" monitoring policy.

<u>API EXAMPLE</u>

method: POST

url:

https://10.122.254.141/api/node/mo/uni/fabric/moncommon/snmpsrc-fab-snmp-source.json

payload

{"snmpSrc":{"attributes":{"dn":"uni/fabric/moncommon/snmpsrc-fab-snmp-source","name":"fab-snmp-source","incl":"events,audit,faults","rn":"snmpsrc-fab-snmp-source","status":"created"},"children":[{"snmpRsDestGroup":{"attributes":{"tDn":"uni/fabric/snmpgroup-fab-snmp-destGrp","status":"created"},"children":[]}}]}}

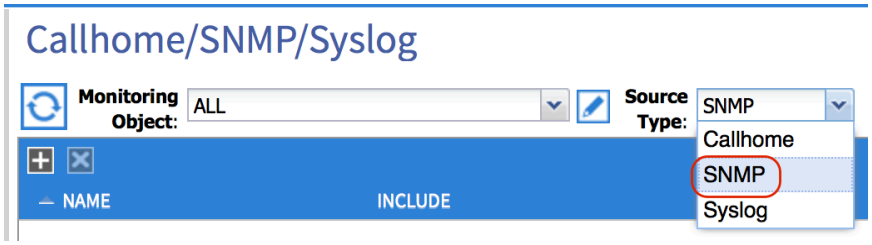
Task 6.2: Configure FABRIC > ACCESS POLICIES to send SNMP TRAPS to SNMP Trap Destinations.

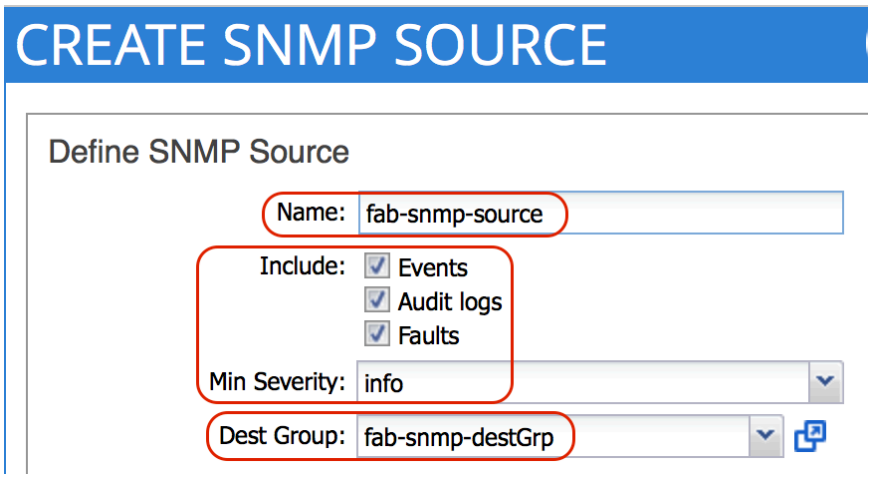
Access policies govern the operation of interfaces that provide external access to the fabric. The system provides default access policies. Access policies enable configuring various functions or protocols. Administrators who have fabric administrator privileges can create new access policies according to their requirements. The APIC enables administrators to select the pods, leaf switches, and interfaces to which they will apply access policies.

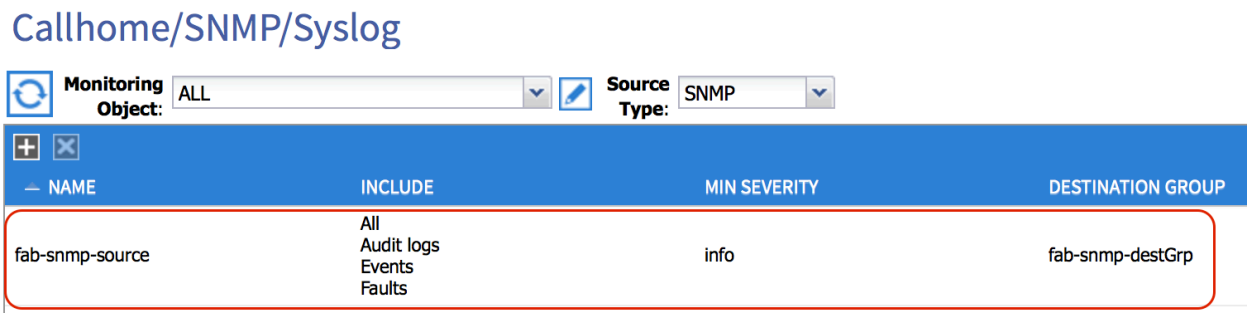
Fabric SNMP Sources need to be configured in the **DEFAULT** monitoring policies configured in the **Access Policies** configuration. Use the GUI to configure the **DEFAULT** SNMP monitoring policies. Use the API Inspector to capture the API POST information from this configuration.

Task 6.2.1 Use the GUI to a configure the "**DEFAULT**" monitoring policy for SNMP. For this task, use the admin user "admin" and the password "Aci123bc".

- On the menu bar, choose **FABRIC > ACCESS POLICIES**. In the Navigation pane, expand **MONITORING POLICIES**.
- Expand **default**
- Select "**Callhome/SNMP/Syslog**"
- In the "Callhome/SNMP/Syslog" Work Pane, **Select** the **SOURCE TYPE "SNMP"** from the Source Type drop down list.
- Click on the " **+** " sign to **CREATE SNMP SOURCE**. In the Create SNMP Source dialog box, perform the following actions:
    - Enter **Name** (fab-snmp-source)
    - For **Include** (Check boxes for Events, Audit logs, and Faults)
    - Select **Min Severity** (information)
    - Select **Dest Group** (fab-snmp-destGrp)
    - Click **SUBMIT**







Using the **APIC API** Inspector, this API Example was captured from the POST request to create SNMP Source for the "**DEFAULT**" monitoring policy for SNMP. You can use this APIC Example and use POSTMAN REST Client to create SNMP Source for the "DEFAULT" monitoring policy.

API EXAMPLE

method: POST

url:

https://10.122.254.141/api/node/mo/uni/infra/moninfra-default/snmpsrc-fab-snmp-source.json

payload

{"snmpSrc":{"attributes":{"dn":"uni/infra/moninfra-default/snmpsrc-fab-snmp-source","name":"fab-snmp-source","incl":"events,audit,faults","rn":"snmpsrc-fab-snmp-source","status":"created"},"children":[{"snmpRsDestGroup":{"attributes":{"tDn":"uni/fabric/snmpgroup-fab-snmp-destGrp","status":"created"},"children":[]}}]}}

response: {"imdata":[]}

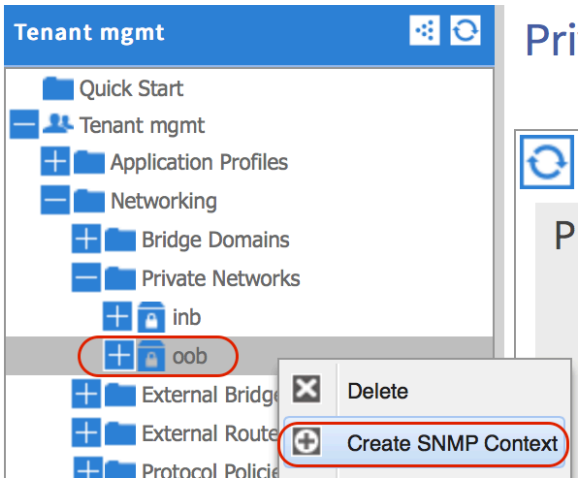
7 Configure SNMP Context for Tenant Management

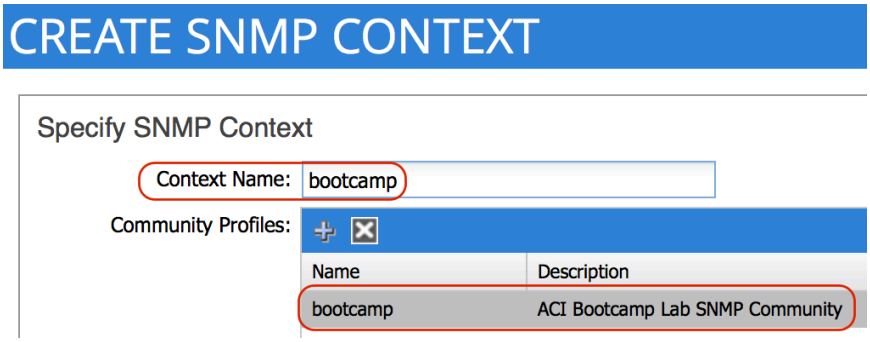
For this lab section, you will configure SNMP Context for Tenant Management OOB Network in your designated ACI fabric. The SNMP Context is a SNMP Version 3 feature. We are configuring SNMP Version 2c. The lab has you configure SNMP Context anyway for configuration awareness.

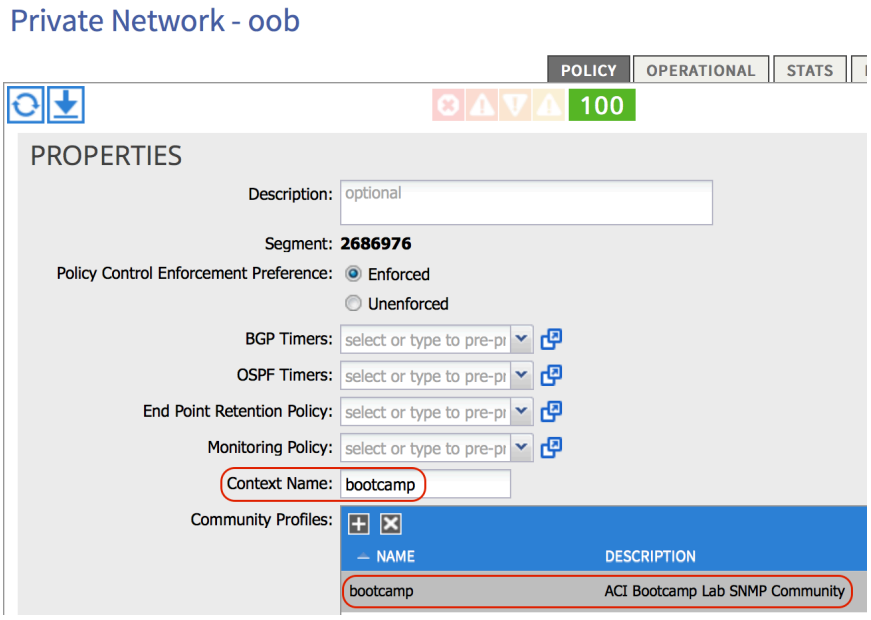
The SNMP context profile, which enables you to specify a context to monitor with a community profile. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

Use the GUI to configure the **SNMP CONTEXT**. Use the API Inspector to capture the API POST information from this configuration. For this task, use the admin user "admin" and the password "Aci123bc".

- On the menu bar, choose **TENANTS > MGMT**. In the Navigation pane, expand **Networking**.
- Expand **Private Networks**
- Select "**oob**"
- Right Click and Click **CREATE SNMP CONTEXT**. In the Create SNMP CONTEXT dialog box, perform the following actions:
  - Enter **Context Name** (bootcamp)
  - Click on the " **+** " sign to **ADD COMMUNITY PROFILES**. In the Community Profiles Table, perform the following actions:
  - Enter **Name** (bootcamp)
  - Click **UPDATE**
- Click **SUBMIT**







Using the **APIC API** Inspector, this API Example was captured from the POST request to create SNMP Context for Tenant Management OOB Network. You can use this APIC Example and use POSTMAN REST Client to create SNMP Context for Tenant Management OOB Network.

<u>API EXAMPLE</u>

method: POST

url:

https://10.122.254.141/api/node/mo/uni/tn-mgmt/ctx-oob/snmpctx.json

payload

{"snmpCtxP":{"attributes":{"dn":"uni/tn-mgmt/ctx-oob/snmpctx","name":"bootcamp","rn":"snmpctx","status":"created"},"children":[{"snmpCommunityP":{"attributes":{"dn":"uni/tn-mgmt/ctx-oob/snmpctx/community-bootcamp","name":"bootcamp","descr":"ACI Bootcamp Lab SNMP Community","rn":"community-bootcamp","status":"created"},"children":[]}}]}}

8 Troubleshooting SNMP Policy for the ACI Fabric

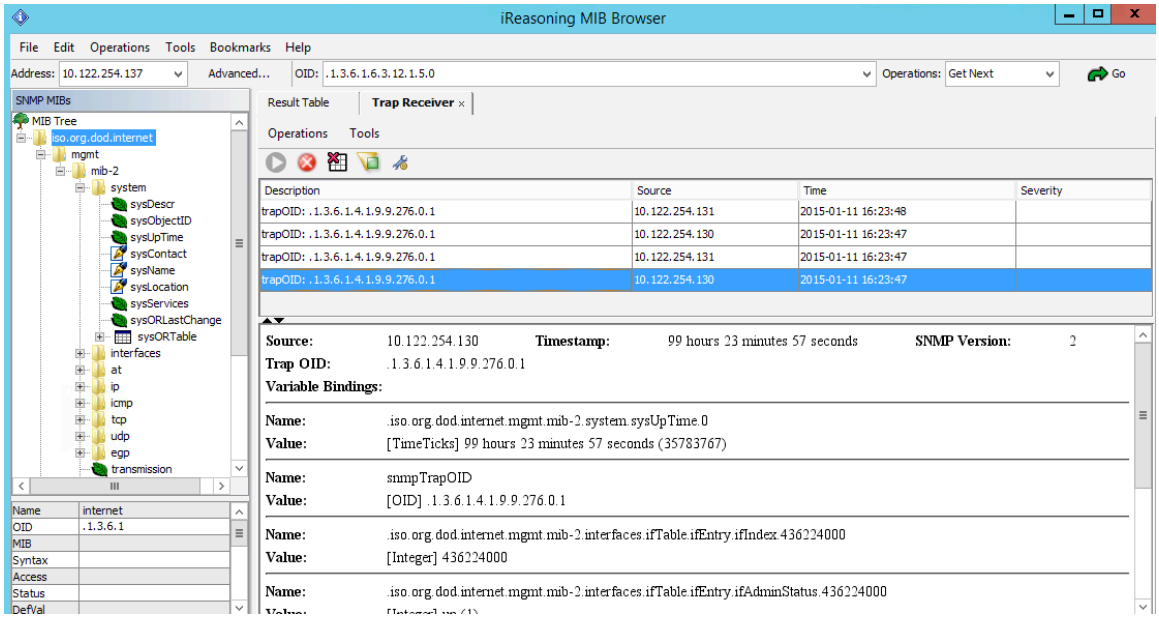
For this lab section, you will verify the configuration of SNMP Services for Leaf\Spine Switches in your designated ACI fabric. This section will provide references for CLI commands and tools that may be helpful in troubleshooting the configuration and application of the SNMP policies for Leaf\Spine Switches in your designated ACI fabric.

This lab section will:

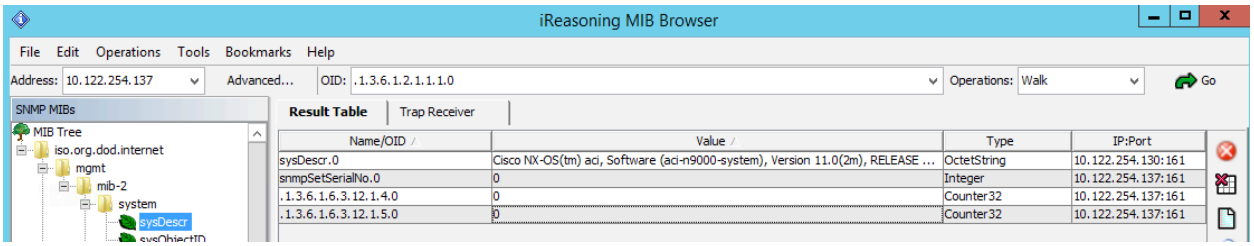
- **Verify SNMP TRAPS are being sent from the Cisco ACI Fabric System & being received by the SNMP Server.**
- **Perform a SNMP Walk against Node Switches from the SNMP MIB Browser.**
- **Verify configuration of SNMP on APIC Controllers and Leaf\Spine Node Switches.**

Note: The examples given in this section of the lab are not totally inclusive. These are just some examples that I have gathered while troubleshooting SNMP Services for the ACI Fabric.

- **Access the Console of your SNMP Server to verify SNMP TRAPS are being sent from the Cisco ACI Fabric System & being received by the SNMP Server.**
    - Remote desktop to your Application Server and review the iReasoning MIB Browser Console. You may need to initiate an error in your Fabric Nodes to trigger an SNMP Trap. (for example, disable an interface that is UP State.)



- **Perform a SNMP Walk against Node Switches from the SNMP MIB Browser.**
  - Remote desktop to your Application Server and review the iReasoning MIB Browser Console. Perform an SNMP Walk with any of the loaded MIBs against Leaf\Spine Switches in your designated Fabric



  - Perform an SNMP Walk to test and verify the **SNMP Context** configuration. The following examples are command line tests that can be used on a MAC OSX client or Windows Client. Remember, if you use a personal device, the IP address of your device needs to be added to the Client Profile of the Fabric's SNMP Policy.

    ⇒ **MAC OSX Command Syntax for SNMP CONTEXT verification**

    - **snmpwalk -v2c -c bootcamp@bootcamp -On 10.122.254.128 ipAddressTable**

```
tdeleon$ snmpwalk -v2c -c bootcamp@bootcamp -On 10.122.254.128 ipAddressTable
.1.3.6.1.2.1.4.34.1.3.1.4.10.122.254.128 = INTEGER: 83886080
.1.3.6.1.2.1.4.34.1.4.1.4.10.122.254.128 = INTEGER: unicast(1)
.1.3.6.1.2.1.4.34.1.5.1.4.10.122.254.128 = OID:
.1.3.6.1.2.1.4.32.1.83886080.1.4.10.122.254.0.24
.1.3.6.1.2.1.4.34.1.6.1.4.10.122.254.128 = INTEGER: manual(2)
.1.3.6.1.2.1.4.34.1.7.1.4.10.122.254.128 = INTEGER: preferred(1)
.1.3.6.1.2.1.4.34.1.8.1.4.10.122.254.128 = Timeticks: (356775) 0:59:27.75
.1.3.6.1.2.1.4.34.1.9.1.4.10.122.254.128 = Timeticks: (356775) 0:59:27.75
.1.3.6.1.2.1.4.34.1.10.1.4.10.122.254.128 = INTEGER: active(1)
.1.3.6.1.2.1.4.34.1.11.1.4.10.122.254.128 = INTEGER: nonVolatile(3)
```

    - **snmpwalk -v2c -c bootcamp@bootcamp -Of 10.122.254.128 ipAddressTable**

```
tdeleon$ snmpwalk -v2c -c bootcamp@bootcamp -Of 10.122.254.128 ipAddressTable
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressIfIndex.ipv4."10.122.254.128" =
INTEGER: 83886080
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressType.ipv4."10.122.254.128" =
INTEGER: unicast(1)
```

```
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressPrefix.ipv4."10.122.254.128" =
OID: .iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressPrefixTable.ipAddressPrefixEntry.83886080.ipv4."10.122.254.0".2
4
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressOrigin.ipv4."10.122.254.128" =
INTEGER: manual(2)
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressStatus.ipv4."10.122.254.128" =
INTEGER: preferred(1)
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressCreated.ipv4."10.122.254.128" =
Timeticks: (356849) 0:59:28.49
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressLastChanged.ipv4."10.122.254.128"
= Timeticks: (356849) 0:59:28.49
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressRowStatus.ipv4."10.122.254.128" =
INTEGER: active(1)
.iso.org.dod.internet.mgmt.mib-
2.ip.ipAddressTable.ipAddressEntry.ipAddressStorageType.ipv4."10.122.254.128"
= INTEGER: nonVolatile(3)
```

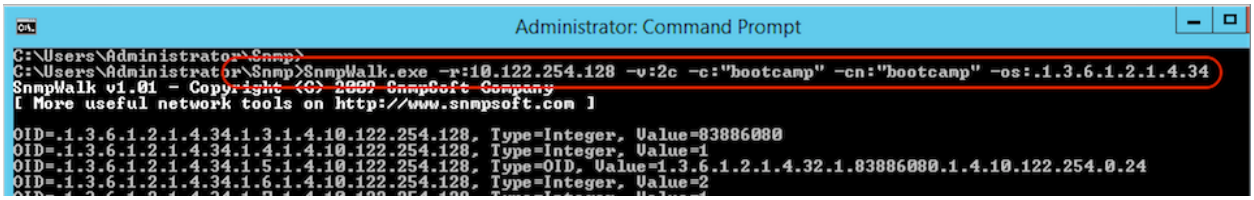
⇒ **Windows Command Syntax for SNMP CONTEXT verification**
You can use your ACI-Px-Server to run this Command line verification test.

- **SnmpWalk.exe -r:10.122.254.128 -v:2c -c:"bootcamp" -cn:"bootcamp" -os:.1.3.6.1.2.1.4.34**

For this lab:

- From your Windows Server, Open a COMMAND PROMPT
- Change directory into SNMP directory. From the Administrator User Directory, "cd snmp"
- Type the SNMP WALK command -> "SnmpWalk.exe -r:10.122.254.128 -v:2c -c:"bootcamp" -cn:"bootcamp" -os:.1.3.6.1.2.1.4.34"

Note: substitute "10.122.254.128" with a Node Switch in your designated Fabric for this lab.



- **Verify configuration of SNMP on APIC Controllers and Leaf\Spine Node Switches.**

  - Verify configuration of SNMP on APIC Controllers.

    ⇒ **CLI Commands**

    ⇒ cat /aci/tenants/mgmt/security-policies/out-of-band-contracts/summary

    ⇒ cat /aci/tenants/mgmt/security-policies/filters/summary

    ⇒ cat /aci/tenants/mgmt/node-management-epgs/default/out-of-band/default/summary

    ⇒ cat /aci/admin/external-data-collectors/monitoring-destinations/snmp/*/snmp-trap-destinations/summary

    ⇒ cat /aci/fabric/fabric-policies/pod-policies/policies/snmp/summary

    ⇒ cat /aci/fabric/fabric-policies/pod-policies/policies/snmp/*/summary

    ⇒ cat /aci/fabric/fabric-policies/pod-policies/policies/snmp/*/client-group-policies/*/*/summary

    ⇒ cat /aci/fabric/fabric-policies/pod-policies/policy-groups/summary

    ⇒ cat /aci/fabric/access-policies/monitoring-policies/default/callhome-snmp-syslog/all/snmp*/summary

    ⇒ cat /aci/fabric/fabric-policies/pod-policies/pod-selector-default-all/summary

    ⇒ cat /aci/fabric/fabric-policies/monitoring-policies/monitoring-policy-default/callhome-snmp-syslog/all/snmp*/summary

    ⇒ cat /aci/fabric/fabric-policies/monitoring-policies/common-policy/callhome-snmp-syslog/snmp/*/summary

    ⇒ moquery -c snmpGroup

    ⇒ moquery -c snmpTrapDest

    ⇒ moquery -c snmpRtDestGroup

    ⇒ moquery -c snmpPol

    ⇒ moquery -c snmpClientGrpP

    ⇒ moquery -c snmpCommunityP

    ⇒ moquery -c snmpRtSnmpPol

    ⇒ moquery -c snmpClientP

    ⇒ moquery -c snmpRsEpg

    ⇒ moquery -c snmpSrc

    ⇒ moquery -c snmpCtxP

- **Visore**
  - ⇒ **snmpGroup** - The SNMP destination group, which contains information needed to send traps or informs to a set of destinations.. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.
  - ⇒ **snmpTrapDest** - A destination to which traps and informs are sent.
  - ⇒ **snmpRtDestGroup** - A target relation to SNMP destination group. This group contains information needed to send traps or informs to a set of destinations
  - ⇒ **snmpPol** - The SNMP policy, which enables you to monitor client group, v3 user, and/or community SNMP policies. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.
  - ⇒ **snmpClientGrpP** - A client group, which is a group of client IP addresses that allows SNMP access to routers or switches.
  - ⇒ **snmpCommunityP** - The SNMP community profile, which enables access to the router or switch statistics for monitoring. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.
  - ⇒ **snmpRtSnmpPol** - A target relation to an SNMP policy that contains site information and general protocol configuration parameters. Note that this relation is an internal object.
  - ⇒ **snmpClientP** - The client profile information.
  - ⇒ **snmpRsEpg** - A source relation to the endpoint group VRF through which the clients can connect. The VRF is an in-band or out-of-band management endpoint.
  - ⇒ **snmpSrc** - The SNMP source profile, which determines the fault information, severity level, and destination for sending messages to the SNMP destination. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.
  - ⇒ **snmpCtxP** - The SNMP context profile, which enables you to specify a context to monitor with a community profile. SNMP is an application-layer protocol that provides a message format for

communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

- **REST API**
  - ⇒ **/api/node/class/snmpGroup.xml?**
  - ⇒ **/api/node/mo/uni/fabric/snmpgroup-fab-snmp-destGrp.xml?query-target=children**
  - ⇒ **/api/node/class/snmpPol.xml?**
  - ⇒ **/api/node/mo/uni/fabric/snmppol-fab-snmp.xml?query-target=children**
  - ⇒ **/api/node/mo/uni/fabric/snmppol-fab-snmp/clgrp-fab-snmpClients.xml?query-target=children**
  - ⇒ **/api/node/class/snmpSrc.xml?**
  - ⇒ **/api/node/class/snmpCtxP.xml?**

- **Verify configuration of SNMP on Leaf\Spine Switches.**
  - ⇒ **CLI Commands**
  - ⇒ (bash) route -n
  - ⇒ (bash) show snmp
  - ⇒ (bash) show snmp summary
  - ⇒ (bash) show snmp community
  - ⇒ (bash) show snmp context
  - ⇒ (bash) show snmp host
  - ⇒ (bash) show snmp internal globals
  - ⇒ (bash) show snmp internal dump-internal-log
  - ⇒ (bash) show snmp summary
  - ⇒ (bash) **netstat -lun output | grep 161**
  - ⇒ (bash) **ps -aux | grep snmp**
    # Verify SNMPD is running
  - ⇒ vsh -c "show snmp"
  - ⇒ vsh -c "show snmp | grep SNMP"
  - ⇒ vsh -c "show snmp community"
  - ⇒ vsh -c "show snmp context"
  - ⇒ show snmp internal oid ?
  - ⇒ **vsh -c "show snmp internal oid dump-internal-log"**
    # Dumps OID Log to File
    (root) iptables –L # Verify IP Rules

Verify SNMP Traps

(root)leaf1# **tcpdump -i eth0 -f port 162**

```
fab2-leaf1# tcpdump -i eth0 -f port 162
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
16:26:42.934279 IP fab2-leaf1-oob.cisco.com.41135 > aci-p1-
server.cisco.com.snmp-trap:  C=bootcamp V2Trap(171)
system.sysUpTime.0=35932280 S:1.1.4.1.0=E:cisco.9.276.0.1
interfaces.ifTable.ifEntry.ifIndex.436240384=436240384
interfaces.ifTable.ifEntry.ifAdminStatus.436240384=2
interfaces.ifTable.ifEntry.ifOperStatus.436240384=2
31.1.1.1.1.436240384="eth1/9" interfaces.ifTable.ifEntry.ifType.436240384=6
16:26:43.034556 IP fab2-leaf1-oob.cisco.com.41135 > aci-p1-
server.cisco.com.snmp-trap:  C=bootcamp V2Trap(171)
system.sysUpTime.0=35932291 S:1.1.4.1.0=E:cisco.9.276.0.1
interfaces.ifTable.ifEntry.ifIndex.436240384=436240384
interfaces.ifTable.ifEntry.ifAdminStatus.436240384=2
interfaces.ifTable.ifEntry.ifOperStatus.436240384=2
31.1.1.1.1.436240384="eth1/9" interfaces.ifTable.ifEntry.ifType.436240384=6
16:26:50.843864 IP fab2-leaf1-oob.cisco.com.41135 > aci-p1-
server.cisco.com.snmp-trap:  C=bootcamp V2Trap(171)
system.sysUpTime.0=35933071 S:1.1.4.1.0=E:cisco.9.276.0.2
interfaces.ifTable.ifEntry.ifIndex.436240384=436240384
interfaces.ifTable.ifEntry.ifAdminStatus.436240384=1
interfaces.ifTable.ifEntry.ifOperStatus.436240384=1
31.1.1.1.1.436240384="eth1/9" interfaces.ifTable.ifEntry.ifType.436240384=6
```

Verify SNMP Requests(Get, Get Next, Get Bulk, Get Subtree, Walk, Set)

(root)leaf1# **tcpdump -i eth0 -f port 161**

```
fab2-leaf1# tcpdump -i eth0 -f port 161
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 65535 bytes
16:27:42.658460 IP aci-p1-server.cisco.com.50855 > fab2-leaf1-
oob.cisco.com.snmp:  C=bootcamp GetRequest(29)  S:12.1.5.0
16:27:42.659086 IP fab2-leaf1-oob.cisco.com.snmp > aci-p1-
server.cisco.com.50855:  C=bootcamp GetResponse(30)  S:12.1.5.0=0
16:28:00.640319 IP aci-p1-server.cisco.com.50856 > fab2-leaf1-
oob.cisco.com.snmp:  C=bootcamp GetNextRequest(23)  .iso.org.dod.internet
16:28:00.641180 IP fab2-leaf1-oob.cisco.com.snmp > aci-p1-
server.cisco.com.50856:  C=bootcamp GetResponse(36)  ip.28.1.2.83886080=65535
16:28:00.641648 IP aci-p1-server.cisco.com.50856 > fab2-leaf1-
oob.cisco.com.snmp:  C=bootcamp GetNextRequest(33)  ip.28.1.2.83886080
16:28:00.641913 IP fab2-leaf1-oob.cisco.com.snmp > aci-p1-
server.cisco.com.50856:  C=bootcamp GetResponse(34)  ip.28.1.3.83886080=1
```

9 Debug Output, Screenshots, and other Reference Information

Reference Material:

- **Cisco APIC Troubleshooting Guide - Troubleshooting Tools and Methodology**
  http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/troubleshooting/b_APIC_Troubleshooting/b_APIC_Troubleshooting_chapter_01.html#d2744e629a1635
- **Cisco Application Centric Infrastructure MIB Quick Reference**
  http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/mib/guide/b_Cisco_ACI_MIB_Quick_Reference.html
- **ACI MIB Support List**
  http://www.cisco.com/c/dam/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/mib/list/mib-support.html
- **SNMP Object Navigator**
  http://tools.cisco.com/Support/SNMP/do/BrowseOID.do?local=en
- **Help! I've configured SNMP on the fabric but the OID I'm trying to query for says SNMP No Such Object or No Such Instance.**
  https://techzone.cisco.com/t5/Application-Centric/Help-I-ve-configured-SNMP-on-the-fabric-but-the-OID-I-m-trying/ta-p/760310
- **Cisco APIC Faults, Events, and System Messages Management Guide**
  http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/faults/guide/b_APIC_Faults_Errors.html
- **Cisco ACI System Messages Reference Guide**
  http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci_syslog/About.html
- **ACI System Messages**
  http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/syslog/guide/aci_syslog/ACI_SysMsg.pdf
- **rfc3411** - An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks
- **rfc5343** - Simple Network Management Protocol (SNMP) Context EngineID Discovery

APIC CLI EXAMPLES

```
admin@fab2-apic1:~> cat /aci/admin/external-data-collectors/monitoring-
destinations/snmp/*/snmp-trap-destinations/summary
snmp-trap-destinations:
host           port  name  management-epg
-------------  ----  ----  ---------------------
10.122.254.77  162         tenants/mgmt/
                           node-management-epgs/
                           default/out-of-band/
                           default
```

```
admin@fab2-apic1:~> cat /aci/fabric/fabric-policies/pod-
policies/policies/snmp/summary
snmp:
name      admin-state  location                       contact
description
--------  -----------  ----------------------------  ------------  ---------
----------------
default   disabled
fab-snmp  enabled      Cisco Systems, North Carolina  Robert Hurst  ACI
Bootcamp Lab for SNMP
```

```
admin@fab2-apic1:~> cat /aci/fabric/fabric-policies/pod-
policies/policies/snmp/*/summary
# snmp-policy
name         : fab-snmp
description  : ACI Bootcamp Lab for SNMP
admin-state  : enabled
contact      : Robert Hurst
location     : Cisco Systems, North Carolina
ownerkey     :
ownertag     :

client-group-policies:
name             description                associated-management-
                                            epg
---------------  -------------------------  ----------------------
fab-snmpClients  ACI Bootcamp Lab for SNMP  tenants/mgmt/
                                            node-management-epgs/
                                            default/out-of-band/
```

```
default
```

VISORE EXAMPLES

snmpPol	
adminSt	enabled
childAction	
contact	Robert Hurst
descr	ACI Bootcamp Lab for SNMP
dn	uni/fabric/snmppol-fab-snmp
lcOwn	local
loc	Cisco Systems, North Carolina
modTs	2015-01-11T16:51:25.751-04:00
monPolDn	uni/fabric/monfab-default
name	fab-snmp
ownerKey	
ownerTag	
status	
uid	15374

snmpTrapDest	
childAction	
descr	
dn	uni/fabric/snmpgroup-fab-snmp-destGrp/trapdest-10.122.254.77-port-162
epgDn	uni/tn-mgmt/mgmtp-default/oob-default
host	10.122.254.77
lcOwn	local
modTs	2015-01-11T17:02:26.317-04:00
monPolDn	uni/fabric/monfab-default
name	
notifT	traps
port	162
secName	bootcamp
status	
uid	15374
v3SecLvl	noauth
ver	v2c
vrfName	

snmpClientGrpP	
childAction	
descr	ACI Bootcamp Lab for SNMP
dn	uni/fabric/snmppol-fab-snmp/clgrp-fab-snmpClients
epgDn	uni/tn-mgmt/mgmtp-default/oob-default
lcOwn	local
modTs	2015-01-11T15:03:19.111-04:00
name	fab-snmpClients
status	
uid	15374

snmpCtxP	
childAction	
dn	uni/tn-mgmt/ctx-oob/snmpctx
lcOwn	local
modTs	2015-01-11T16:14:21.925-04:00
name	bootcamp
status	
uid	15374

snmpClientP	
addr	10.122.254.77
childAction	
dn	uni/fabric/snmppol-fab-snmp/clgrp-fab-snmpClients/client-[10.122.254.77]
lcOwn	local
modTs	2015-01-11T15:03:19.111-04:00
name	aci-p1-server
status	
uid	15374

LEAF\SPINE SWITCH CLI EXAMPLES

```
fab2-leaf1# netstat -lun output | grep 161
udp        0      0 0.0.0.0:161             0.0.0.0:*
udp6       0      0 :::161                  :::*


fab2-leaf1# ps -aux | grep snmp
root      4265  0.0  1.0 1750732 166588 ?      Ssl  Jan07   0:52
/isan/bin/snmpd -f -s -d udp:161 udp6:161 tcp:161


fab2-leaf1# vsh -c "show snmp | grep SNMP"
0 SNMP packets input
   0 Bad SNMP versions
2 SNMP packets output
         SNMP USERS
SNMP protocol : Enabled



fab2-leaf1# show snmp summary

Admin State : enabled, running (pid:4265)

Local SNMP engineID: [Hex] 80000009037C69F6105BF9
                     [Dec] 128:000:000:009:003:124:105:246:016:091:249

---------------------------------------------------------------------
Community           Context             Status
---------------------------------------------------------------------
bootcamp            bootcamp            ok

---------------------------------------------------------------------
User                Authentication      Privacy             Status
---------------------------------------------------------------------

---------------------------------------------------------------------
Context             VRF                 Status
---------------------------------------------------------------------
bootcamp            management          ok

---------------------------------------------------------------------
Client              VRF                 Status
---------------------------------------------------------------------
10.122.254.77       management          ok

-------------------------------------------------------------------------------
---
Host                Port Ver  Level          SecName        VRF
Status
-------------------------------------------------------------------------------
---
10.122.254.77       162  v2c  noauth         bootcamp       management
```

```
fab2-leaf1# show snmp host
-----------------------------------------------------------------
Host                            Port Version  Level  Type   SecName
-----------------------------------------------------------------
10.122.254.77                   162  v2c      noauth trap   bootcamp
Use VRF: management


fab2-leaf1# show snmp context
Context             VRF
_______             ___
bootcamp            management
```

Model References

Class snmp:TrapDest (CONCRETE)

Class ID:1691
Class Label: SNMP Trap Destination
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin]
Read Access: [admin, ops]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: true, HasStats: false, HasFaults: true, HasHealth: true]

A destination to which traps and informs are sent.

Class snmp:Pol (CONCRETE)

Class ID:4571
Class Label: SNMP Policy
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin, fabric-protocol-mgmt]
Read Access: [admin, fabric-connectivity-l1, fabric-connectivity-l2, fabric-connectivity-l3, fabric-equipment, fabric-protocol-l1, fabric-protocol-l2, fabric-protocol-l3, fabric-protocol-mgmt]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: true, HasStats: false, HasFaults: false, HasHealth: true]

The SNMP policy, which enables you to monitor client group, v3 user, and/or community SNMP policies. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

Class snmp:ClientGrpP (CONCRETE)

Class ID:4579
Class Label: SNMP Client Group Profile
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin, fabric-protocol-mgmt]
Read Access: [admin, fabric-protocol-mgmt]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

A client group, which is a group of client IP addresses that allows SNMP access to routers or switches.

Class snmp:Group (CONCRETE)

Class ID:1692
Class Label: SNMP Monitoring Destination Group
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin]
Read Access: [admin, ops]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: true, HasStats: false, HasFaults: false, HasHealth: true]

The SNMP destination group, which contains information needed to send traps or informs to a set of destinations.. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

Class snmp:RtDestGroup (CONCRETE)

Class ID:1690
Class Label: SNMP Source
Encrypted: false - Exportable: false - Persistent: true - Configurable: false
Relationship Type: explicit
Relationship Cardinality: n-to-1
Relationship From: snmp:Src
Relationship From Rel: snmp:RsDestGroup
Relationship To: snmp:Group
Relationship To Rel: snmp:RtDestGroup
Enforceable: true
Resolvable: true
Write Access: [NON CONFIGURABLE]
Read Access: [admin, ops]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

A target relation to SNMP destination group. This group contains information needed to send traps or informs to a set of destinations

Class snmp:RtSnmpPol (CONCRETE)

Class ID:913
Class Label: POD Policy Group
Encrypted: false - Exportable: false - Persistent: true - Configurable: false
Relationship Type: named
Relationship Cardinality: n-to-1
Relationship From: fabric:PodPGrp
Relationship From Rel: fabric:RsSnmpPol
Relationship To: snmp:Pol
Relationship To Rel: snmp:RtSnmpPol
Enforceable: true
Resolvable: true
Write Access: [NON CONFIGURABLE]
Read Access: [admin, fabric-connectivity-l1, fabric-connectivity-l2, fabric-connectivity-l3, fabric-equipment, fabric-protocol-l1, fabric-protocol-l2, fabric-protocol-l3, fabric-protocol-mgmt]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

A target relation to an SNMP policy that contains site information and general protocol configuration parameters. Note that this relation is an internal object.

Class snmp:CommunityP (CONCRETE)

Class ID:4575
Class Label: SNMP Community
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin, fabric-protocol-mgmt]
Read Access: [admin, fabric-protocol-mgmt]
Possible Semantic Scopes: Fabric, EPG,
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

The SNMP community profile, which enables access to the router or switch statistics for monitoring. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

Class snmp:ClientP (CONCRETE)

Class ID:4585
Class Label: Client Entry
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin, fabric-protocol-mgmt]
Read Access: [admin, fabric-protocol-mgmt]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

The client profile information.

Class snmp:RsEpg (CONCRETE)

Class ID:4580
Class Label: Attachable Target Group
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Relationship Type: explicit
Relationship Cardinality: n-to-1
Relationship From: snmp:ClientGrpP
Relationship From Rel: snmp:RsEpg
Relationship To: fv:ATg
Relationship To Rel: fv:RtEpg
Enforceable: false
Resolvable: false
Write Access: [admin, fabric-protocol-mgmt]
Read Access: [admin, fabric-protocol-mgmt]
Semantic Scope: Fabric
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

A source relation to the endpoint group VRF through which the clients can connect. The VRF is an in-band or out-of-band management endpoint.

Class snmp:Src (CONCRETE)

Class ID:1688
Class Label: SNMP Source
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [admin]
Read Access: [admin, ops]
Possible Semantic Scopes: Fabric, Infra, EPG, Common,
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: true, HasStats: false, HasFaults: false, HasHealth: true]

The SNMP source profile, which determines the fault information, severity level, and destination for sending messages to the SNMP destination. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network .

Class snmp:CtxP (CONCRETE)

Class ID:4587
Class Label: SNMP Context Profile
Encrypted: false - Exportable: true - Persistent: true - Configurable: true
Write Access: [access-protocol-mgmt, admin, fabric-protocol-mgmt, tenant-connectivity-l3]
Read Access: [access-protocol-mgmt, admin, fabric-protocol-mgmt, tenant-connectivity-l3]
Semantic Scope: EPG
Semantic Scope Evaluation Rule: Parent
Monitoring Policy Source: Parent
Monitoring Flags : [IsObservable: false, HasStats: false, HasFaults: false, HasHealth: false]

The SNMP context profile, which enables you to specify a context to monitor with a community profile. SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language used for the monitoring and management of devices in a network.

End of Document