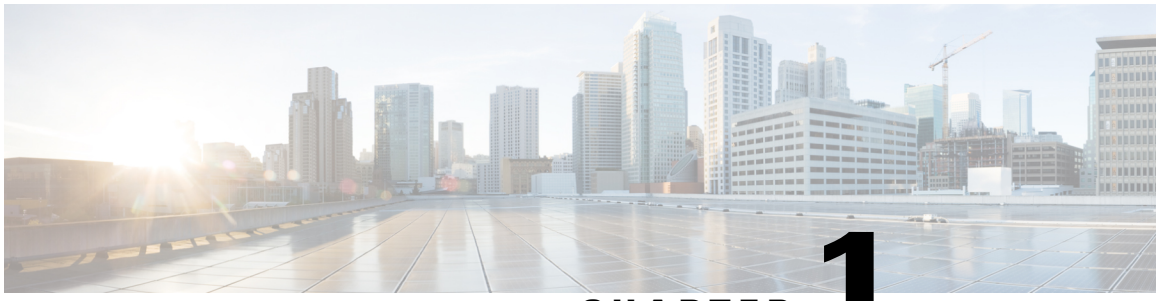
SNMP Configuration Guide, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)

First Published: August 06, 2012

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



CHAPTER 1

Simple Network Management Protocol

Simple Network Management Protocol (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 that is used for monitoring and managing devices in a network.

This module discusses how to enable an SNMP agent on a Cisco device and how to control the sending of SNMP notifications from the agent. For information about using SNMP management systems, see the appropriate documentation for your network management system (NMS) application.

For a complete description of the device monitoring commands mentioned in this document, see the Cisco Network Management Command Reference. To locate documentation of other commands that appear in this document, use the Cisco IOS Master Command List or search online.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for SNMP

Not all Cisco platforms are supported on the features described in this module. Use Cisco Feature Navigator to find information about platform support and Cisco software image support.

Information About Configuring SNMP Support

Components of SNMP

The Simple Network Management Protocol (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 monitoring and managing devices in a network.

The SNMP framework has the following components, which are described in the following sections:

SNMP Operations

SNMP applications perform the following operations to retrieve data, modify SNMP object variables, and send notifications:

SNMP Get

The Simple Network Management Protocol (SNMP) GET operation is performed by an Network Management Server (NMS) to retrieve SNMP object variables. There are three types of GET operations:

- GET—Retrieves the exact object instance from the SNMP agent.
- GETNEXT—Retrieves the next object variable, which is a lexicographical successor to the specified variable.
- GETBULK—Retrieves a large amount of object variable data, without the need for repeated GETNEXT operations.

SNMP SET

The Simple Network Management Protocol (SNMP) SET operation is performed by a Network Management Server (NMS) to modify the value of an object variable.

SNMP Notifications

A key feature of SNMP is its capability to generate unsolicited notifications from an SNMP agent.

Traps and Informs

Unsolicited (asynchronous) notifications can be generated as traps or inform requests (informs). Traps are messages alerting the SNMP manager to a condition on the network. Informs are traps that include a request for confirmation of receipt from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighboring device, or other significant events.

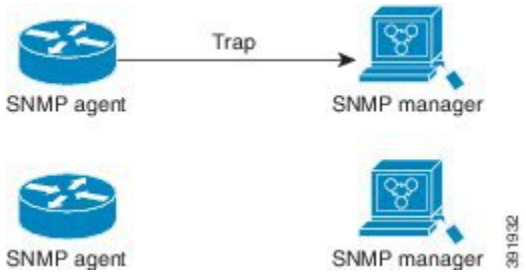
Traps are less reliable than informs because the receiver does not send an acknowledgment when it receives a trap. The sender does not know if the trap was received. An SNMP manager that receives an inform, acknowledges the message with an SNMP response protocol data unit (PDU). If the sender never receives a response, the inform can be sent again. Thus, informs are more likely to reach their intended destination.

Traps are often preferred even though they are less reliable because informs consume more resources in the device and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform must be held in memory until a response is received or the request times out. Also, traps are sent only once, whereas an inform may be resent several times. The retries increase traffic and contribute to higher overhead on the network. Use of traps and informs requires a trade-off between reliability and resources. If it is important that the SNMP manager receives every notification, use informs, but if traffic volume or memory usage are concerns and receipt of every notification is not required, use traps.

The following figures illustrate the differences between traps and informs.

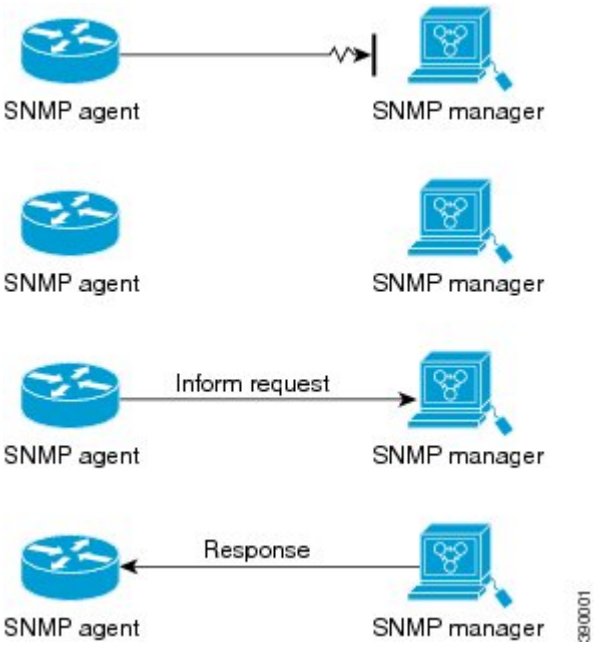
The figure below shows that an agent successfully sends a trap to an SNMP manager. Although the manager receives the trap, it does not send an acknowledgment. The agent has no way of knowing that the trap reached its destination.

Figure 1: Trap Successfully Sent to SNMP Manager



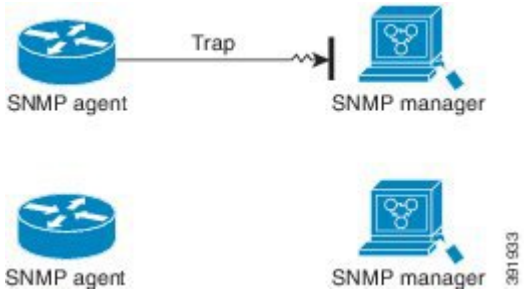
In the figure below, the agent successfully sends an inform to the manager. When the manager receives the inform, a response is sent to the agent and the agent knows that the inform reached its destination. Notice that in this example, the traffic generated is twice as much as in the interaction shown in the table above.

Figure 2: Inform Request Successfully Sent to SNMP Manager



The figure below shows an agent sending a trap to a manager that the manager does not receive. The agent has no way of knowing that the trap did not reach its destination. The manager never receives the trap because traps are not resent.

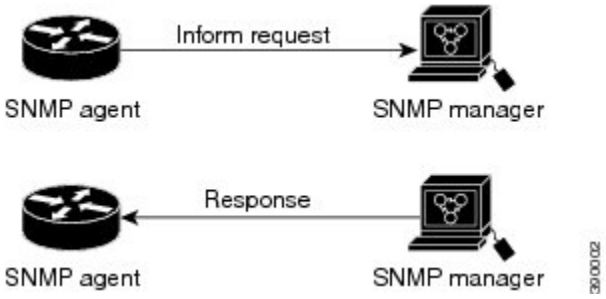
Figure 3: Trap Unsuccessfully Sent to SNMP Manager



The figure below shows an agent sending an inform to a manager that does not reach the manager. Because the manager did not receive the inform, it does not send a response. After a period of time, the agent resends the inform. The manager receives the inform from the second transmission and replies. In this example, more

traffic is generated than in the scenario shown in the table above but the notification reaches the SNMP manager.

Figure 4: Inform Unsuccessfully Sent to SNMP Manager



Versions of SNMP

The Cisco IOS software supports the following versions of SNMP:

- SNMPv1—Simple Network Management Protocol: a full Internet standard, defined in RFC 1157. (RFC 1157 replaces the earlier versions that were published as RFC 1067 and RFC 1098.) Security is based on community strings.
- SNMPv2c—The community string-based Administrative Framework for SNMPv2. SNMPv2c (the "c" is for "community") is an experimental Internet protocol defined in RFC 1901, RFC 1905, and RFC 1906. SNMPv2c is an update of the protocol operations and data types of SNMPv2p (SNMPv2 Classic) and uses the community-based security model of SNMPv1.
- SNMPv3—Version 3 of SNMP. SNMPv3 is an interoperable standards-based protocol defined in RFCs 3413 to 3415. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network.

The security features provided in SNMPv3 are as follows:

- Message integrity—Ensuring that a packet has not been tampered with in transit.
- Authentication—Determining that the message is from a valid source.
- Encryption—Scrambling the contents of a packet to prevent it from being learned by an unauthorized source.

Both SNMPv1 and SNMPv2c use a community-based form of security. The community of SNMP managers able to access the agent MIB is defined by a community string.

SNMPv2c support includes a bulk retrieval mechanism and detailed error message reporting to management stations. The bulk retrieval mechanism supports the retrieval of tables and large quantities of information, minimizing the number of round trips required. The SNMPv2c improved error handling support includes expanded error codes that distinguish different types of errors; these conditions are reported through a single error code in SNMPv1. The following three types of exceptions are also reported: no such object, no such instance, and end of MIB view.

SNMPv3 is a security model in which an authentication strategy is set up for a user and the group in which the user resides. A security level is the permitted level of security within a security model. A combination of

a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Three security models are available: SNMPv1, SNMPv2c, and SNMPv3. The table below lists the combinations of security models and levels and their meanings.

Table 1: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community String	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
v3	authPriv	MD5 or SHA	Data Encryption Standard (DES)	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard.

Note SNMPv2p (SNMPv2 Classic) is not supported in Cisco IOS Release 11.2 and later releases. SNMPv2c replaces the Party-based Administrative and Security Framework of SNMPv2p with a Community-based Administrative Framework. SNMPv2c retained the bulk retrieval and error handling capabilities of SNMPv2p.

You must configure an SNMP agent to use the version of SNMP supported by the management station. An agent can communicate with multiple managers. You can configure the Cisco IOS software to support communications with one management station using the SNMPv1 protocol, one using the SNMPv2c protocol, and another using SNMPv3.

SNMPv3 supports RFCs 1901 to 1908, 2104, 2206, 2213, 2214, and 2271 to 2275. For additional information about SNMPv3, see RFC 2570, *Introduction to Version 3 of the Internet-standard Network Management Framework* (this is not a standards document).

How to Configure SNMP Support

There is no specific command to enable SNMP. The first **snmp-server** command that you enter enables supported versions of SNMP. All other configurations are optional.

Configuring System Information

You can set the system contact, location, and serial number of the SNMP agent so that these descriptions can be accessed through the configuration file. Although the configuration steps described in this section are optional, configuring the basic information is recommended because it may be useful when troubleshooting your configuration. In addition, the first **snmp-server** command that you issue enables SNMP on the device.

Perform this task as needed.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server contact** *text*
4. **snmp-server location** *text*
5. **snmp-server chassis-id** *number*
6. end
7. **show snmp contact**
8. **show snmp location**
9. **show snmp chassis**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**snmp-server contact** *text* **Example:** `Device(config)# snmp-server contact NameOne`	Sets the system contact string.
Step 4	**snmp-server location** *text* **Example:** `Device(config)# snmp-server location LocationOne`	Sets the system location string.
Step 5	**snmp-server chassis-id** *number* **Example:** `Device(config)# snmp-server chassis-id 015A619T`	Sets the system serial number.
Step 6	end **Example:** `Device(config)# end`	Exits global configuration mode.
Step 7	**show snmp contact** **Example:** `Device# show snmp contact`	(Optional) Displays the contact strings configured for the system.
Step 8	**show snmp location** **Example:** `Device# show snmp location`	(Optional) Displays the location string configured for the system.
Step 9	**show snmp chassis** **Example:** `Device# show snmp chassis`	(Optional) Displays the system serial number.

Enabling the SNMP Agent Shutdown Mechanism

Using SNMP packets, a network management tool can send messages to users on virtual terminals and on the console. This facility operates in a similar fashion to the **send** EXEC command; however, the SNMP request that causes the message to be issued to the users also specifies the action to be taken after the message is delivered. One possible action is a shutdown request. After a system is shut down, typically it is reloaded.

Because the ability to cause a reload from the network is a powerful feature, it is protected by the **snmp-server system-shutdown** global configuration command. If you do not issue this command, the shutdown mechanism is not enabled.

Perform this task to enable the SNMP agent shutdown mechanism.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server system-shutdown**
4. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server system-shutdown** **Example:** `Device(config)# snmp-server system-shutdown`	Enables system shutdown using the SNMP message reload feature.
Step 4	end **Example:** `Device(config)# end`	Exits global configuration mode.

Defining the Maximum SNMP Agent Packet Size

You can define the maximum packet size permitted when the SNMP agent is receiving a request or generating a reply.

Perform this task to set the maximum permitted packet size.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server packetsize** *byte-count*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server packetsize** *byte-count* **Example:** `Device(config)# snmp-server packetsize 512`	Establishes the maximum packet size.
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Limiting the Number of TFTP Servers Used via SNMP

You can limit the number of TFTP servers used for saving and loading configuration files via SNMP by using an access list. Limiting the use of TFTP servers in this way conserves system resources and centralizes the operation for manageability.

Perform this task to limit the number of TFTP servers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server tftp-server-list** *number*
4. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server tftp-server-list** *number* **Example:** `Device(config)# snmp-server tftp-server-list 12`	Limits the number of TFTP servers used for configuration file copies via SNMP to the servers in an access list.
Step 4	end **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Troubleshooting Tips

An alternative to using the ifAlias value for the identification of interfaces across reboots is to use the cciDescr object in the Cisco Circuit Interface MIB (CISCO-CIRCUIT-INTERFACE-MIB.my). This MIB object can be used only for circuit-based interfaces such as ATM or Frame Relay interfaces. Cisco IOS feature FTS-731 introduced the Circuit Interface Identification Persistence for the Simple Network Management Protocol (SNMP), which maintains the user-defined name of the circuit (defined in the cciDescr object) across reboots and allows consistent identification of circuit-based interfaces.

Configuring SNMP Versions 1 and 2

When you configure SNMP versions 1 and 2, you can optionally create or modify views for community strings to limit which MIB objects an SNMP manager can access.

Perform the following tasks when configuring SNMP version 1 or version 2.

Creating or Modifying an SNMP View Record

You can assign views to community strings to limit which MIB objects an SNMP manager can access. You can use a predefined view or create your own view. If you are using a predefined view or no view at all, skip this task.

Perform this task to create or modify an SNMP view record.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server view** *view-name oid-tree* {**included** | **excluded**}
4. **no snmp-server view** *view-name oid-tree* {**included** | **excluded**}
5. **end**
6. show snmp view

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server view** *view-name oid-tree* {**included** | **excluded**} **Example:** `Device(config)# snmp-server view mib2 mib-2 included`	Creates a view record. • In this example, the mib2 view that includes all objects in the MIB-II subtree is created. **Note** You can use this command multiple times to create the same view record. If a view record for the same OID value is created multiple times, the latest entry of the object identifier takes precedence.

	Command or Action	Purpose
Step 4	**no snmp-server view** *view-name oid-tree* {**included** | **excluded**} **Example:** `Device(config)# no snmp-server view mib2 mib-2 included`	Removes a server view.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 6	show snmp view **Example:** `Device# show snmp view`	(Optional) Displays a view of the MIBs associated with SNMP.

Creating or Modifying Access Control for an SNMP Community

Use an SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to regulate access to the agent on the device. Optionally, you can specify one or more of the following characteristics associated with the string:

- An access list of IP addresses of the SNMP managers that are permitted to use the community string to gain access to the agent.
- A MIB view, which defines the subset of all MIB objects accessible to the given community.
- Read and write or read-only permission for the MIB objects accessible to the community.

Perform this task to create or modify a community string.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**ipv6** *nacl*] [*access-list-number*]
4. **no snmp-server community** *string*
5. **end**
6. **show snmp community**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**ipv6** *nacl*] [*access-list-number*] **Example:** `Device(config)# snmp-server community comaccess ro 4`	Defines the community access string. • You can configure one or more community strings.
Step 4	**no snmp-server community** *string* **Example:** `Device(config)# no snmp-server community comaccess`	Removes the community string from the configuration.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 6	**show snmp community** **Example:** `Device# show snmp community`	(Optional) Displays the community access strings configured for the system.

Configuring a Recipient of an SNMP Trap Operation

To enable multiple hosts, you must issue a separate **snmp-server host** command for each host. You can specify multiple notification types in the command for each host.

When multiple **snmp-server host** commands are given for the same host and kind of notification (trap or inform), each succeeding command overwrites the previous command. Only the last **snmp-server host** command will be in effect. For example, if you enter an **snmp-server host inform** command for a host and

then enter another **snmp-server host inform** command for the same host, the second command will replace the first.

The **snmp-server host** command is used in conjunction with the **snmp-server enable** command. Use the **snmp-server enable** command to specify which SNMP notifications are sent globally. For a host to receive most notifications, at least one **snmp-server enable** command and the **snmp-server host** command for that host must be enabled.

Some notification types cannot be controlled with the **snmp-server enable** command. For example, some notification types are always enabled and others are enabled by a different command. For example, the linkUpDown notifications are controlled by the **snmp trap link-status** interface configuration command. These notification types do not require an **snmp-server enable** command.

A *notification-type* option's availability depends on the device type and Cisco IOS software features supported on the device. For example, the Cisco IOS software does not support the envmon notification type. To see what notification types are available on your system, use the command help (?) at the end of the **snmp-server host** command.

Perform this task to configure the recipient of an SNMP trap operation.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-id* [**traps** | **informs**][**version** {**1**| **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port-number*] [*notification-type*]
4. **end**
5. **show snmp host**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server host** *host-id* [**traps**	**informs**][**version** {**1**| **2c**	**3** [**auth**	**noauth**	**priv**]}] *community-string* [**udp-port** *port-number*] [*notification-type*] **Example:** `Device(config)# snmp-server host 172.16.1.27 version 2c public`	Specifies whether you want the SNMP notifications sent as traps or informs, the version of SNMP to use, the security level of the notifications (for SNMPv3), and the recipient (host) of the notifications.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 5	**show snmp host** **Example:** `Device# show snmp host`	(Optional) Displays the SNMP notifications sent as traps, the version of SNMP, and the host IP address of the notifications.

Disabling the SNMP Agent

Perform this task to disable any version of an SNMP agent.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **no snmp-server**
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**no snmp-server** **Example:** `Device(config)# no snmp-server`	Disables SNMP agent operation.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Configuration Examples for SNMP Support

Example: Configuring SNMPv1 Support

The following example shows how to enable SNMPv1. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string named public. This configuration does not cause the router to send traps.

```
Device(config)# snmp-server community public
```

The following example shows how to permit SNMP access to all objects with read-only permission using the community string named public. The router also will send BGP traps to the hosts 172.16.1.111 and 172.16.1.33 using SNMPv1. The community string named public is sent with the traps.

```
Device(config)# snmp-server community public
Device(config)# snmp-server enable traps bgp
Device(config)# snmp-server host 172.16.1.111 version 1 public
Device(config)# snmp-server host 172.16.1.33 public
```

The following example shows how to send the SNMP and Cisco environmental monitor enterprise-specific traps to address 172.30.2.160:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host 172.30.2.160 public snmp envmon
```

The following example shows how to enable the router to send all traps to the host example.com using the community string public:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host example.com public
```

The following example shows a configuration in which no traps are sent to a host. The BGP traps are enabled for all hosts, but only the OSPF traps are enabled to be sent to a host.

```
Device(config)# snmp-server enable traps bgp
Device(config)# snmp-server host host1 public ospf
```

The following example shows how to enable a router to send all informs to the host example.com using the community string named public:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host example.com informs version 2c public
```

The following example shows how to enable the SNMP manager and set the session timeout to a value greater than the default:

```
Device(config)# snmp-server manager
Device(config)# snmp-server manager session-timeout 1000
```

The following example shows how to enable the SNMP manager to access all objects with read-only permissions. The user is specified as *abcd* and the authentication password is *abcdpasswd*. To obtain the automatically generated default local engine ID, use the **show snmp engineID** command.

```
Device(config)# snmp-server view readview internet included
Device(config)# snmp-server view readview iso included
Device(config)# snmp-server group group1 v3 noauth read readview
Device(config)# snmp-server user abcd group1 v3 auth md5 abcdpasswd
```

Example: Show SNMP View

The following example shows the SNMP view for the system OID tree:

```
Device> enable
Device# configure terminal
Device(config)# snmp-server view test system included
Device(config)# end
Device# show snmp view

test system - included nonvolatile active
cac_view pimMIB - included read-only active
cac_view msdpMIB - included read-only active
cac_view interfaces - included read-only active
cac_view ip - included read-only active
cac_view ospf - included read-only active
.
.
.
v1default iso - included permanent active
v1default internet - included permanent active
v1default snmpUsmMIB - excluded permanent active
v1default snmpVacmMIB - excluded permanent active
v1default snmpCommunityMIB - excluded permanent active
v1default ciscoIpTapMIB - excluded permanent active
v1default ciscoMgmt.395 - excluded permanent active
v1default ciscoTap2MIB - excluded permanent active
.
.
.
```

Example Configuring SNMP Community Access Strings

The following example shows the community access strings configured to enable access to the SNMP manager:

```
Device> enable
Device# configure terminal
Device(config)# snmp-server community public ro
Device(config)# snmp-server community private rw
Device(config)# end
Device# show snmp community

Community name: private
```

```
Community Index: private
Community SecurityName: private
storage-type: nonvolatile active
Community name: public
Community Index: public
Community SecurityName: public
storage-type: nonvolatile active
```

Example Configuring Host Information

The following example shows the host information configured for SNMP notifications:

```
Device> enable
Device# configure terminal
Device(config)# snmp-server host 10.2.28.1 inform version 2c public
Device(config)# end
Device# show snmp host

Notification host: 10.2.28.1 udp-port: 162   type: inform
user: public    security model: v2c
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*

Standard/RFC	Title
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

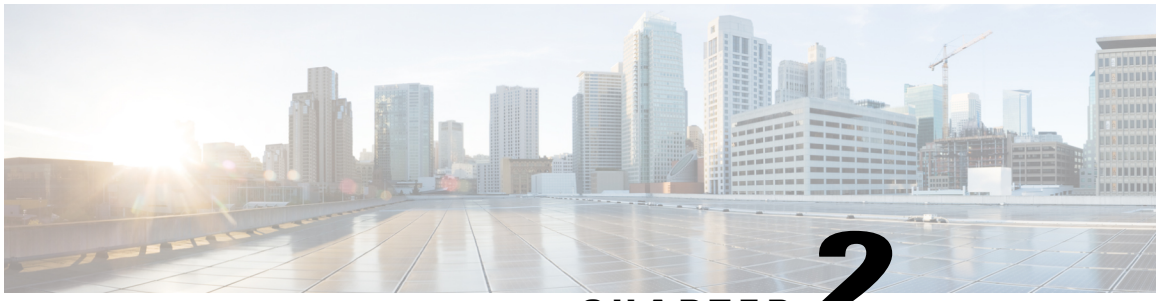
Feature Information for Simple Network Management Protocol

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 2: Feature Information for Simple Network Management Protocol

Feature Name	Releases	Feature Information
SNMP (Simple Network Management Protocol)		The Simple Network Management Protocol (SNMP) feature provides an application-layer protocol that facilitates the exchange of management information between network devices. SNMP is part of the TCP/IP protocol suite. SNMP enables network administrators to manage network performance, find and solve network problems, and plan for network growth.



CHAPTER 2

SNMP Manager

The Simple Network Management Protocol (SNMP) Manager feature allows a device to serve as an SNMP manager. As an SNMP manager, the device can send SNMP requests to agents and receive SNMP responses and notifications from agents. When the SNMP manager process is enabled, the device can query other SNMP agents and process incoming SNMP traps.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about SNMP Manager

Overview

The SNMP manager is a system that controls and monitors the activities of network hosts using SNMP. The most common managing system is a network management system (NMS). The term NMS can be applied either to a dedicated device used for network management or to the applications used on such a device. Several network management applications are available for use with SNMP and range from simple command-line applications to applications that use GUIs, such as the CiscoWorks2000 products.

The SNMP manager feature allows a device to act as a network management station--an SNMP client. As an SNMP manager, the device can send SNMP requests to agents and receive SNMP responses and notifications from agents. When the SNMP manager process is enabled, the device can query other SNMP agents and process incoming SNMP traps.

Most network security policies assume that devices will accept SNMP requests, send SNMP responses, and send SNMP notifications.

With the SNMP manager functionality enabled, the device may also send SNMP requests, receive SNMP responses, and receive SNMP notifications. Your security policy implementation may need to be updated prior to enabling this feature.

SNMP requests typically are sent to UDP port 161. SNMP responses are typically sent from UDP port 161. SNMP notifications are typically sent to UDP port 162.

Sessions are created when the SNMP manager in the device sends SNMP requests, such as informs, to a host or receives SNMP notifications from a host. One session is created for each destination host. If there is no further communication between the device and host within the session timeout period, the session will be deleted.

The device tracks statistics, such as the average round-trip time required to reach the host, for each session. Using the statistics for a session, the SNMP manager in the device can set reasonable timeout periods for future requests, such as informs, for that host. If the session is deleted, all statistics are lost. If another session with the same host is later created, the request timeout value for replies will return to the default value.

Sessions consume memory. A reasonable session timeout value should be large enough that regularly used sessions are not prematurely deleted, yet small enough such that irregularly used or one-time sessions are purged expeditiously.

Security Considerations

Most network security policies assume that the devices will be accepting SNMP requests, sending SNMP responses, and sending SNMP notifications.

With the SNMP manager functionality enabled, the device may also be sending SNMP requests, receiving SNMP responses, and receiving SNMP notifications. Your security policy implementation may need to be updated prior to enabling this feature.

SNMP requests are typically sent to UDP port 161. SNMP responses are typically sent from UDP port 161. SNMP notifications are typically sent to UDP port 162.

How to Configure SNMP Manager

Configuring a Device as an SNMP Manager

Perform this task to enable the SNMP manager process and to set the session timeout value.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server manager**
4. **snmp-server manager session-timeout** *seconds*
5. **end**
6. **show snmp**
7. **show snmp sessions [brief]**
8. **show snmp pending**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server manager** **Example:** `Device(config)# snmp-server manager`	Enables the SNMP manager.
Step 4	**snmp-server manager session-timeout** *seconds* **Example:** `Device(config)# snmp-server manager session-timeout 30`	(Optional) Changes the session timeout value.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 6	**show snmp** **Example:** `Device# show snmp`	(Optional) Displays the status of SNMP communications.
Step 7	**show snmp sessions [brief]** **Example:** `Device# show snmp sessions`	(Optional) Displays the status of SNMP sessions.

	Command or Action	Purpose
Step 8	**show snmp pending** **Example:** `Device# show snmp pending`	(Optional) Displays the current set of pending SNMP requests.

Examples

The following example shows the status of SNMP communications:

```
Device# show snmp

Chassis: 01506199
37 SNMP packets input
    0 Bad SNMP version errors
    4 Unknown community name
    0 Illegal operation for community name supplied
    0 Encoding errors
    24 Number of requested variables
    0 Number of altered variables
    0 Get-request PDUs
    28 Get-next PDUs
    0 Set-request PDUs
78 SNMP packets output
    0 Too big errors (Maximum packet size 1500)
    0 No such name errors
    0 Bad values errors
    0 General errors
    24 Response PDUs
    13 Trap PDUs
SNMP logging: enabled
    Logging to 172.17.58.33.162, 0/10, 13 sent, 0 dropped.
SNMP Manager-role output packets
    4 Get-request PDUs
    4 Get-next PDUs
    6 Get-bulk PDUs
    4 Set-request PDUs
    23 Inform-request PDUs
    30 Timeouts
    0 Drops
SNMP Manager-role input packets
    0 Inform response PDUs
    2 Trap PDUs
    7 Response PDUs
    1 Responses with errors
SNMP informs: enabled
    Informs in flight 0/25 (current/max)
    Logging to 172.17.217.141.162
        4 sent, 0 in-flight, 1 retries, 0 failed, 0 dropped
    Logging to 172.17.58.33.162
        0 sent, 0 in-flight, 0 retries, 0 failed, 0 dropped
```

The following example displays the status of SNMP sessions:

```
Device# show snmp sessions

Destination: 172.17.58.33.162, V2C community: public
  Round-trip-times: 0/0/0 (min/max/last)
  packets output
    0 Gets, 0 GetNexts, 0 GetBulks, 0 Sets, 4 Informs
    0 Timeouts, 0 Drops
  packets input
    0 Traps, 0 Informs, 0 Responses (0 errors)
Destination: 172.17.217.141.162, V2C community: public, Expires in 575 secs
  Round-trip-times: 1/1/1 (min/max/last)
```

```
  packets output
    0 Gets, 0 GetNexts, 0 GetBulks, 0 Sets, 4 Informs
    0 Timeouts, 0 Drops
 packets input
    0 Traps, 0 Informs, 4 Responses (0 errors)
```

The following example shows the current set of pending SNMP requests:

```
Device# show snmp pending

req id: 47, dest: 172.17.58.33.161, V2C community: public, Expires in 5 secs
req id: 49, dest: 172.17.58.33.161, V2C community: public, Expires in 6 secs
req id: 51, dest: 172.17.58.33.161, V2C community: public, Expires in 6 secs
req id: 53, dest: 172.17.58.33.161, V2C community: public, Expires in 8 secs
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*

Standard/RFC	Title
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

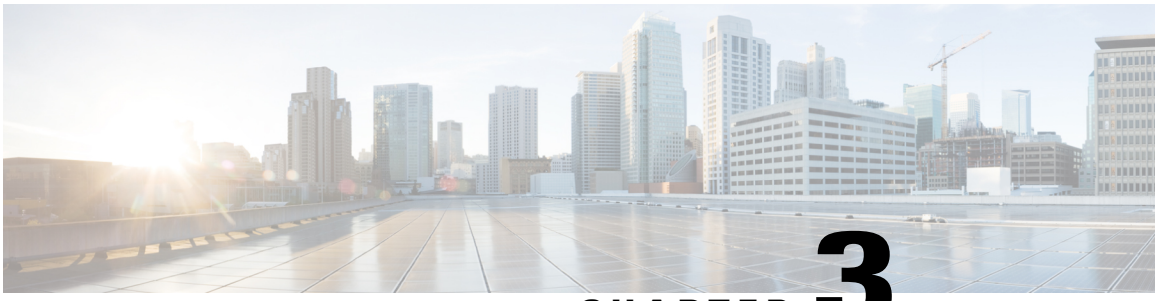
Feature Information for SNMP Manager

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 3: Feature Information for SNMP Manager

Feature Name	Releases	Feature Information
SNMP Manager	11.3(1) 11.3(1)T 12.0(1) 15.0(1)S	The SNMP Manager feature adds a system that controls and monitors the activities of network hosts using SNMP. The most common managing system is an NMS.



CHAPTER 3

SNMP Diagnostics

The Simple Network Management Protocol (SNMP) diagnostics feature allows you to diagnose configuration problems. The feature can be used to troubleshoot, debug connectivity issues, packet loss, and latency in a LAN environment with the help of Cisco command line interface commands.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about SNMP Diagnostics

SNMP Diagnostics

The SNMP Diagnostics feature adds Cisco command line interface commands to display the object identifiers that are recently requested by the network management system, and to display the SNMP debug messages.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

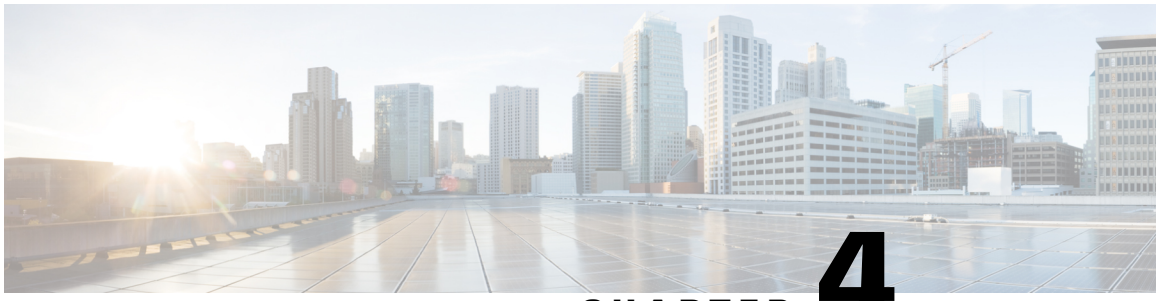
Feature Information for SNMP Diagnostics

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 4: Feature Information for SNMP Diagnostics

Feature Name	Releases	Feature Information
SNMP Diagnostics	12.2(33)SRE 12.2(50)SY 12.4(20)T 15.0(1)S	The SNMP Diagnostics feature adds Cisco command line interface commands to display object identifiers recently requested by the network management system, and to display the SNMP debug messages. The following commands were introduced or modified: **debug**, **snmp detail**, **show snmp stats oid**.



CHAPTER 4

SNMP Trap Simulations

The Simple Network Management Protocol (SNMP) Trap Simulations feature provides information about the command line interface commands to the SNMP manager to verify the reception of notifications in a simulated environment.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About SNMP Trap Simulations

SNMP Trap Simulations

The SNMP Trap Simulations feature introduces the **test snmp trap** commands to verify the reception of SNMP, syslog, and config-copy notifications by the SNMP manager in a simulated scenario.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

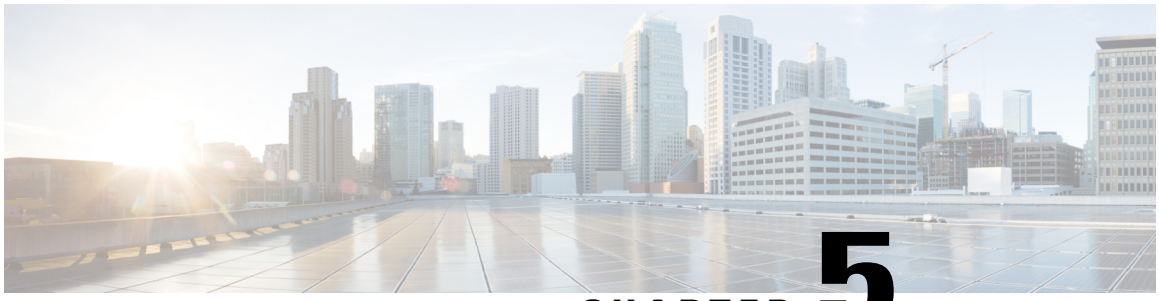
Feature Information for SNMP Trap Simulations

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 5: Feature Information for SNMP Trap Simulations

Feature Name	Releases	Feature Information
SNMP Trap Simulations	12.2(33)SRE 12.2(33)SXI 15.0(1)S	The SNMP Trap Simulations feature introduces the commands to verify the reception of SNMP, syslog, and config-copy notifications by the SNMP manager in a simulated scenario. The following commands were introduced or modified: **test snmp trap**



CHAPTER 5

SNMP Notification Logging

Systems that support Simple Network Management Protocol (SNMP) often need a mechanism for recording notification information as a hedge against lost notifications, whether those are traps or informs that exceed retransmission limits. The Notification Log MIB provides a common infrastructure for other MIBs in the form of a local logging function. The SNMP Notification Logging feature adds Cisco command line interface commands to change the size of the notification log, to set the global ageout value for the log, and to display logging summaries at the command line.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About SNMP Notification Logging

SNMP Notification Logging

Systems that support SNMP often need a mechanism for recording notification information. This mechanism protects against notifications being lost because they exceeded retransmission limits. The Notification Log MIB provides a common infrastructure for other MIBs in the form of a local logging function. The SNMP

Notification Logging feature adds Cisco command line interface commands to change the size of the notification log, to set the global ageout value for the log, and to display logging summaries at the command line. The Notification Log MIB improves notification tracking and provides a central location for tracking all MIBs.

You can globally enable or disable authenticationFailure, linkUp, linkDown, warmStart, and coldStart traps or informs individually. (These traps constitute the "generic traps" defined in RFC 1157.) Note that linkUp and linkDown notifications are enabled by default on specific interfaces but will not be sent unless they are enabled globally.

Note The Notification Log MIB supports notification logging on the default log only.

Benefits

Benefits of using SNMP notification logging are as follows:

- Improves notification tracking.
- Provides a central location for tracking all MIBs.

How to Configure SNMP Notification Logging

Configuring SNMP Notifications

To configure a device to send SNMP traps or informs, perform the tasks described in the following sections:

Note Many snmp-server commands use the keyword **traps** in their command syntax. Unless there is an option within the command to specify either traps or informs, the keyword **traps** should be taken to mean traps, informs, or both. Use the **snmp-server host** command to specify whether you want SNMP notifications to be sent as traps or informs. To use informs, the SNMP manager (also known as the SNMP proxy manager) must be available and enabled on a device. Earlier, the SNMP manager was available only with Cisco IOS PLUS images. However, the SNMP manager is now available with all Cisco software releases that support SNMP. Use Cisco Feature Navigator for information about SNMP manager support for Cisco software releases. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn.

Configuring the Device to Send SNMP Notifications

Perform this task to configure the device to send traps or informs to a host.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server engineID remote** *remote-ip-address remote-engineID*
4. **snmp-server user** *username groupname* [**remote** *host* [**udp-port** *port*] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**access** *access-list*]
5. **snmp-server group** *groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]
6. **snmp-server host** *host* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [*notification-type*]
7. **snmp-server enable traps** [*notification-type* [*notification-options*]]
8. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server engineID remote** *remote-ip-address remote-engineID* **Example:** `Device(config)# snmp-server engineID remote 172.16.20.3 80000009030000B064EFE100`	Specifies the SNMP engine ID and configures the VRF name traps-vrf for SNMP communications with the remote device at 172.16.20.3.
Step 4	**snmp-server user** *username groupname* [**remote** *host* [**udp-port** *port*] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**access** *access-list*] **Example:** `Device(config)# snmp-server user abcd public v3 encrypted auth md5 cisco123`	Configures a local or remote user to an SNMP group. **Note** You cannot configure a remote user for an address without first configuring the engine ID for that remote host. This restriction is imposed in the design of these commands; if you try to configure the user before the host, you will receive a warning message and the command will not be executed. Use the **snmp-server engineid remote** command to specify the engine ID for a remote host.
Step 5	**snmp-server group** *groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configures an SNMP group.

	Command or Action	Purpose
	Example: `Device(config)# snmp-server group GROUP1 v2c auth read viewA write viewA notify viewB`	
Step 6	**snmp-server host** *host* [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [*notification-type*] **Example:** `Device(config)# snmp-server host example.com informs version 3 public`	Specifies whether you want the SNMP notifications sent as traps or informs, the version of SNMP to use, the security level of the notifications (for SNMPv3), and the recipient (host) of the notifications. • The **snmp-server host** command specifies which hosts will receive SNMP notifications, and whether you want the notifications sent as traps or informs.
Step 7	**snmp-server enable traps** [*notification-type* [*notification-options*]] **Example:** `Device(config)# snmp-server enable traps bgp`	Enables sending of traps or informs and specifies the type of notifications to be sent. • If a *notification-type* is not specified, all supported notification are enabled on the device. • To discover which notifications are available on your device, enter the **snmp-server enable traps ?** command. • The **snmp-server enable traps** command globally enables the production mechanism for the specified notification types (such as Border Gateway Protocol [BGP] traps, config traps, entity traps, Hot Standby Device Protocol [HSDP] traps, and so on).
Step 8	end **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.

Changing Notification Operation Values

You can specify a value other than the default for the source interface, message (packet) queue length for each host, or retransmission interval.

Perform this task to change notification operation values as needed.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server trap-source** *interface*
4. **snmp-server queue-length** *length*
5. **snmp-server trap-timeout** *seconds*
6. **snmp-server informs** [**retries** *retries*] [**timeout** *seconds*] [**pending** *pending*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server trap-source** *interface* **Example:** `Device(config)# snmp-server trap-source FastEthernet 2/1`	Sets the IP address for the Fast Ethernet interface in slot2, port 1 as the source for all SNMP notifications.
Step 4	**snmp-server queue-length** *length* **Example:** `Device(config)# snmp-server queue-length 50`	Establishes the message queue length for each notification. • This example shows the queue length set to 50 entries.
Step 5	**snmp-server trap-timeout** *seconds* **Example:** `Device(config)# snmp-server trap-timeout 30`	Defines how often to resend notifications on the retransmission queue.
Step 6	**snmp-server informs** [**retries** *retries*] [**timeout** *seconds*] [**pending** *pending*] **Example:** `Device(config)# snmp-server informs retries 10 timeout 30 pending 100`	Configures inform-specific operation values. • This example sets the maximum number of times to resend an inform, the number of seconds to wait for an acknowledgment before resending, and the maximum number of informs waiting for acknowledgments at any one time.

	Command or Action	Purpose

Controlling Individual RFC 1157 SNMP Traps

Perform this task to enable the authenticationFailure, linkUp, linkDown, warmStart, and coldStart notification types.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server enable traps snmp** [**authentication**] [**linkup**] [**linkdown**] [**warmstart**] [**coldstart**]
4. **interface** *type slot/port*
5. **no snmp-server link-status**
6. **end**
7. **end**
8. **show snmp mib ifmibtraps**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server enable traps snmp** [**authentication**] [**linkup**] [**linkdown**] [**warmstart**] [**coldstart**] **Example:** `Device(config)# snmp-server enable traps snmp`	Enables RFC 1157 generic traps. • When used without any of the optional keywords, enables authenticationFailure, linkUp, linkDown, warmStart, and coldStart traps. • When used with keywords, enables only the trap types specified. For example, to globally enable only linkUp and linkDown SNMP traps or informs for all interfaces, use the **snmp-server enable traps snmp linkup linkdown** form of this command.

	Command or Action	Purpose
Step 4	**interface** *type slot/port* **Example:** `Device(config)# interface FastEthernet 0/0`	Enters interface configuration mode for a specific interface. **Note** To enable SNMP traps for individual interfaces such as Dialer, use the **snmp trap link-status permit duplicates** command in interface configuration mode. For example, to enter dialer interface configuration mode, enter the interface type as dialer.
Step 5	**no snmp-server link-status** **Example:** `Device(config-if)# no snmp-server link-status`	Disables the sending of linkUp and linkDown notifications for all generic interfaces.
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode.
Step 7	**end** **Example:** `Device(config)# end`	Exits global configuration mode and returns to privileged EXEC mode.
Step 8	**show snmp mib ifmibtraps** **Example:** `Device# show snmp mib ifmib traps`	

Examples

The following example shows the status of linkup and linkdown traps for all interfaces configured for the system:

```
Device# show snmp mib ifmib traps

ifDescr   ifindex  TrapStatus
---------------------------------
FastEthernet  3/6 14  enabled
FastEthernet  3/19 27  enabled
GigabitEthernet  5/1 57  enabled
unrouted VLAN  1005 73  disabled
FastEthernet  3/4 12  enabled
FastEthernet  3/39 47  enabled
FastEthernet  3/28 36  enabled
FastEthernet  3/48 56  enabled
unrouted VLAN  1003 74  disabled
FastEthernet  3/2 10  enabled
Tunnel   0 66  enabled
SPAN RP Interface  64  disabled
```

```
Tunnel   10 67  enabled
FastEthernet  3/44 52  enabled
GigabitEthernet  1/3 3  enabled
FastEthernet  3/11 19  enabled
FastEthernet  3/46 54  enabled
GigabitEthernet  1/1 1  enabled
FastEthernet  3/13 21  enabled
unrouted VLAN  1 70  disabled
GigabitEthernet  1/4 4  enabled
FastEthernet  3/9 17  enabled
FastEthernet  3/16 24  enabled
FastEthernet  3/43 51  enabled
```

Configuring SNMP Notification Log Options

Perform this task to configure SNMP notification log options. These options allow you to control the log size and timing values. The SNMP log can become very large and long, if left unmodified.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib notification-log default**
4. **snmp mib notification-log globalageout** *seconds*
5. **snmp mib notification-log globalsize** *size*
6. **end**
7. **show snmp mib notification-log**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib notification-log default** **Example:** `Device(config)# snmp mib notification-log default`	Creates an unnamed SNMP notification log.
Step 4	**snmp mib notification-log globalageout** *seconds* **Example:** `Device(config)# snmp mib notification-log globalageout 20`	Sets the maximum amount of time for which the SNMP notification log entries remain in the system memory.

	Command or Action	Purpose
		• In this example, the system is configured to delete entries in the SNMP notification log that were logged more than 20 minutes ago.
Step 5	**snmp mib notification-log globalsize** *size* **Example:** `Device(config)# snmp mib notification-log globalsize 600`	Sets the maximum number of entries that can be stored in all SNMP notification logs.
Step 6	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 7	**show snmp mib notification-log** **Example:** `Device# show snmp mib notification-log`	Displays information about the state of the local SNMP notification logging.

Examples

This example shows information about the state of local SNMP notification logging:

```
Device# show snmp mib notification-log

GlobalAgeout 20, GlobalEntryLimit 600
Total Notifications logged in all logs 0
Log Name"", Log entry Limit 600, Notifications logged 0
Logging status enabled
Created by cli
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module

Related Topic	Document Title
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*

Standard/RFC	Title
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

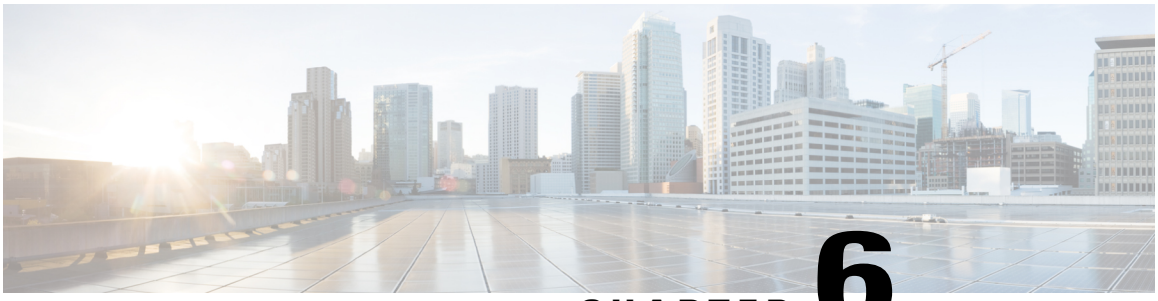
Feature Information for SNMP Notification Logging

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 6: Feature Information for SNMP Notification Logging

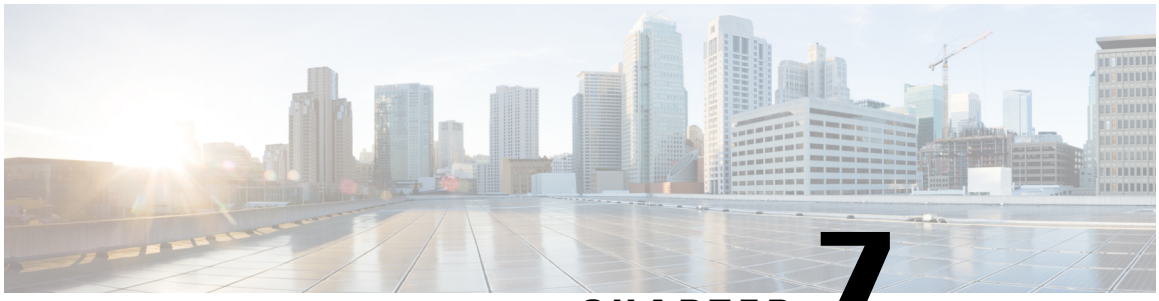
Feature Name	Releases	Feature Information
SNMP Notification Logging	12.0(22)S 12.2(13)T	The SNMP Notification Logging feature adds Cisco command line interface commands to change the size of the notification log, set the global ageout value for the log, and display logging summaries at the command line.



CHAPTER 6

Memory Pool—SNMP Notification Support

This feature adds Cisco command line interface commands to enable Simple Network Management Protocol (SNMP) notifications for the Cisco Enhanced Memory Pool MIB (CISCO-ENHANCED-MEMPOOL-MIB).



CHAPTER 7

SNMP Inform Request

The Simple Network Management Protocol (SNMP) Inform Requests feature allows devices to send inform requests to SNMP managers.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About SNMP Inform Requests

SNMP Inform Request

The SNMP Inform Request feature supports sending inform requests. SNMP asynchronous notifications are usually sent as SNMP traps.

Traps are less reliable than informs because an acknowledgment is not sent from the receiving end when a trap is received; however, an SNMP manager that receives an inform acknowledges the message with an SNMP response PDU. If the sender does not receive a response for an inform, the inform can be sent again.

How to Configure SNMP Inform Requests

Configuring Devices to Send Traps

Perform the following task to configure the device to send traps to a host in global configuration mode:

SUMMARY STEPS

1. enable
2. **configure terminal**
3. **snmp-server host** *host*[**version {1|2c}**]*community-string*[**udp-port** *port*][*notification-type*]
4. **snmp-server enable traps**[*notification-type*] [*notification-option*]
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server host** *host*[**version {1|2c}**]*community-string*[**udp-port** *port*][*notification-type*] **Example:** `Device(config)# snmp-server host 10.10.10.10 version 1 public udp-port 2012`	Specifies the recipient of the trap message.
Step 4	**snmp-server enable traps**[*notification-type*] [*notification-option*] **Example:** `Device(config)# snmp-server enable traps alarms 3`	Globally enables the trap production mechanism for the specified traps. **Note** Some traps are not controlled by the **snmp-serverenabletraps** command. These traps are either enabled by default or controlled through other commands. For example, by default, SNMP link traps are sent when an interface goes up or down. For interfaces expected to go up and down during normal usage, such as ISDN interfaces, the output generated by

	Command or Action	Purpose
		these traps may not be useful. Use the **nosnmptrapslink-status** interface configuration command to disable these traps. In order for a host to receive a trap, an **snmp-serverhost** command must be configured for that host, and the trap must be enabled globally through the **snmp-serverenabletraps** command, through a different command, such as **snmptrapslink-status**, or by default.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Changing Inform Operation Values

Perform the following optional task in global configuration mode to change inform operation values:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server informs** [**retries** *retries*] [**timeout** *seconds*] [**pending** *pending*]
4. **snmp-server trap-source** *interface*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server informs** [**retries** *retries*] [**timeout** *seconds*] [**pending** *pending*]	Configures inform-specific operation values. • This example sets the maximum number of times to resend an inform, the number of seconds to wait for an

	Command or Action	Purpose
	Example: `Device(config)# snmp-server informs retries 10 timeout 30 pending 100`	acknowledgment before resending, and the maximum number of informs waiting for acknowledgments at any one time.
Step 4	**snmp-server trap-source** *interface* **Example:** `Device(config)# snmp-server trap-source Gigabitethernet 1/2/1`	This example sets the IP address for the Fast Ethernet interface in slot2, port 1 as the source for all SNMP notifications.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Configuration Examples for SNMP Inform Request

Example: Configuring SNMP Inform Request

The following configuration example shows how to configure the SNMP Inform Request feature for SNMPv1 or SNMPv2:

The following example sends the SNMP and Cisco environmental monitor enterprise-specific traps to address 172.30.2.160:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host 172.30.2.160 public snmp envmon
```

The following example enables the device to send all traps to the host myhost.example.com using the community string public:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host myhost.example.com public
```

The following example will not send traps to any host. The BGP traps are enabled for all hosts, but only the ISDN traps are enabled to be sent to a host.

```
Device(config)# snmp-server enable traps bgp
Device(config)# snmp-server host bob public isdn
```

The following example enables the device to send all inform requests to the host myhost.example.com using the community string public:

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host myhost.example.com informs version 2c public
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

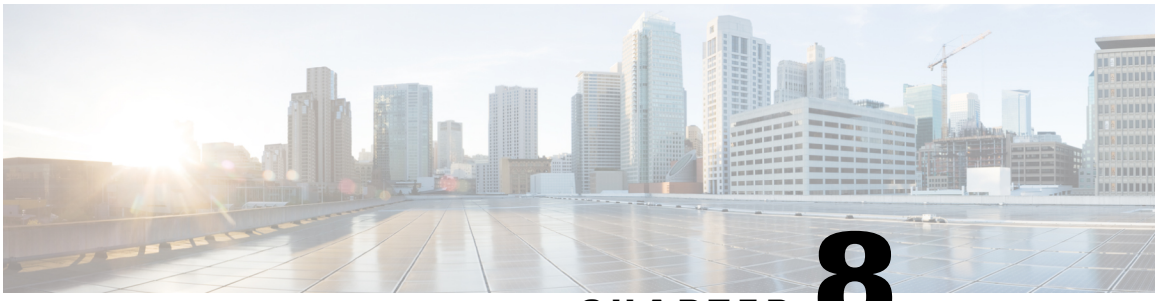
Feature Information for SNMP Inform Request

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 7: Feature Information for SNMP Inform Request

Feature Name	Releases	Feature Information
SNMP Inform Request		The SNMP Inform Request feature supports sending inform requests. SNMP asynchronous notifications are usually sent as SNMP traps. Traps are less reliable than informs because an acknowledgment is not sent from the receiving end when a trap is received; however, an SNMP manager that receives an inform acknowledges the message with an SNMP response PDU. If the sender does not receive a response for an inform, the inform can be sent again.



CHAPTER 8

SNMP Support for VPNs

The Simple Network Management Protocol (SNMP) Support for VPNs feature allows the sending and receiving of SNMP notifications (traps and informs) using VPN routing and forwarding (VRFs) tables. In particular, this feature adds support to Cisco software for the sending and receiving of SNMP notifications (traps and informs) specific to individual VPNs.

The SNMP Support for VPNs feature provides configuration commands that allow users to associate SNMP agents and managers with specific VRFs. The specified VRF is used for the sending of SNMP notifications (traps and informs) and responses between agents and managers. If a VRF is not specified, the default routing table for the VPN is used.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about SNMP Support for VPNs

SNMP Support for VPNs

The SNMP Support for VPNs feature allows SNMP traps and informs to be sent and received using VPN routing/forwarding (VRF) tables. In particular, this feature adds support to Cisco software for sending and receiving SNMP traps and informs that are specific to individual VPNs.

A VPN is a network that provides high connectivity transfers on a shared system with the same usage guidelines as a private network. A VPN can be built on the Internet over IP, Frame Relay, or ATM networks.

A VRF stores per-VPN routing data. It defines the VPN membership of a customer site attached to the network access server (NAS). A VRF consists of an IP routing table, a derived Cisco Express Forwarding table, and guidelines and routing protocol parameters that control the information that is included in the routing table.

The SNMP Support for VPNs feature provides configuration commands that allow users to associate SNMP agents and managers with specific VRFs. The specified VRF is used for sending SNMP traps and informs and responses between agents and managers. If a VRF is not specified, the default routing table for the VPN is used.

The SNMP Support for VPNs feature allows you to configure an SNMP agent to accept only SNMP requests from a certain set of VPNs. With this configuration, service providers can provide network management services to their customers, so that the customers can manage all user VPN devices.

How to Configure SNMP Support for VPNs

Configuring SNMP Support for VPNs

This section describes how to configure SNMP support for VPNs. The SNMP Support for VPNs feature provides configuration commands that allow users to associate SNMP agents and managers with specific VRFs. The specified VRF is used to send SNMP traps and informs and responses between agents and managers. If a VRF is not specified, the default routing table for the VPN is used.

Support for VPNs allows users to configure an SNMP agent to only accept SNMP requests from a certain set of VPNs. With this configuration, providers can provide network management services to their customers who then can manage all user VPN devices.

Note

- This feature is not supported on all Cisco platforms. Use Cisco Feature Navigator to find information about platform support and Cisco software image support.
- Not all MIBs are VPN aware. To list the VPN-aware MIBs, use the **show snmp mib context** command.

Perform this task to configure SNMP support for a specific VPN.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server host** *host-address* [**vrf** *vrf-name*] [**traps** | **informs**] [**version** {**1**| **2c**| **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*]
4. **snmp-server engineID remote** *ip-address* [**udp-port** *udp-port-number*] [**vrf** *vrf-name*] *engineid-string*
5. **end**
6. **show snmp host**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** Perform this task to configure SNMP support for a specific VPN. `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server host** *host-address* [**vrf** *vrf-name*] [**traps**	**informs**] [**version** {**1**| **2c**| **3** [**auth**	**noauth**	**priv**]}] *community-string* [**udp-port** *port*] [*notification-type*] **Example:** `Device(config)# snmp-server host example.com vrf trap-vrf public`	Specifies the recipient of an SNMP notification operation and specifies the VRF table to be used for the sending of SNMP notifications.
Step 4	**snmp-server engineID remote** *ip-address* [**udp-port** *udp-port-number*] [**vrf** *vrf-name*] *engineid-string* **Example:** `Device(config)# snmp-server engineID remote 172.16.20.3 vrf traps-vrf`	Configures a name for the remote SNMP engine on a device when configuring SNMP over a specific VPN for a remote SNMP user.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

	Command or Action	Purpose
Step 6	**show snmp host** **Example:** `Device# show snmp host`	(Optional) Displays the SNMP configuration and verifies that the SNMP Support for VPNs feature is configured properly.

Configuration Example for SNMP Support for VPNs

Example: Configuring SNMP Support for VPNs

In the following example all SNMP notifications are sent to example.com over the VRF named trap-vrf:

```
Device(config)# snmp-server host example.com vrf trap-vrf
```

In the following example the VRF named "traps-vrf" is configured for the remote server 172.16.20.3:

```
Device(config)# snmp-server engineID remote 172.16.20.3 vrf traps-vrf 80000009030000B064EFE100
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*

Standard/RFC	Title
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

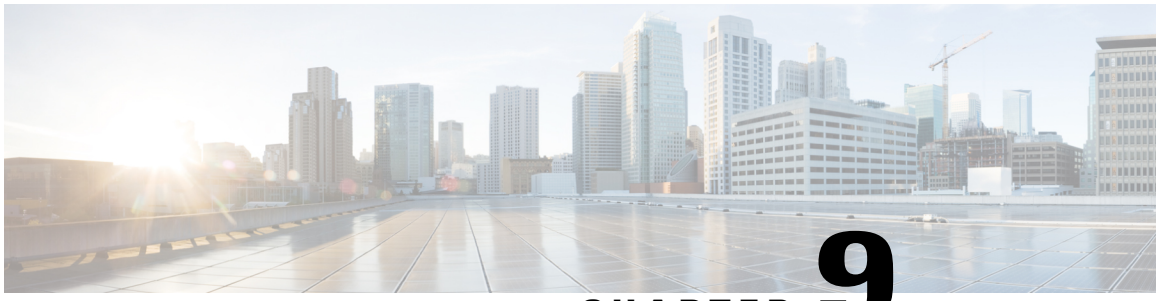
Feature Information for SNMP Support for VPNs

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 8: Feature Information for SNMP Support for VPNs

Feature Name	Releases	Feature Information
SNMP Support for VPNs	12.0(23)S 12.2(2)T 12.2(33)SB 12.2(33)SXH 15.0(1)S Cisco IOS XE Release 3.1.0SG	The SNMP Support for VPNs feature allows SNMP traps and informs to be sent and received using VRF tables. In particular, this feature adds support to the Cisco software for sending and receiving SNMP traps and informs specific to individual VPNs.



CHAPTER 9

SNMP Support over VPNs—Context-Based Access Control

The SNMP Support over VPNs—Context-Based Access Control feature provides the infrastructure for multiple Simple Network Management Protocol (SNMP) context support in Cisco software and VPN-aware MIB infrastructure using the multiple SNMP context support infrastructure.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for SNMP Support over VPNs—Context-Based Access Control

- If you delete an SNMP context using the **no snmp-server context** command, all SNMP instances in that context are deleted.

- Not all MIBs are VPN-aware.

Information About SNMP Support over VPNs—Context-Based Access Control

SNMP Versions and Security

Cisco software supports the following versions of SNMP:

- SNMPv1—Simple Network Management Protocol: a full Internet standard, defined in RFC 1157. (RFC 1157 replaces the earlier versions that were published as RFC 1067 and RFC 1098.) Security is based on community strings.
- SNMPv2c—The community string-based Administrative Framework for SNMPv2. SNMPv2c (the "c" is for "community") is an experimental Internet protocol defined in RFC 1901, RFC 1905, and RFC 1906. SNMPv2c is an update of the protocol operations and data types of SNMPv2p (SNMPv2 Classic) and uses the community-based security model of SNMPv1.

For more information about SNMP Versions, see the " Configuring SNMP Support " module in the *Cisco Network Management Configuration Guide*.

SNMPv1 or SNMPv2 Security

Cisco IOS software supports the following versions of SNMP:

- SNMPv1—Simple Network Management Protocol: a full Internet standard, defined in RFC 1157. (RFC 1157 replaces the earlier versions that were published as RFC 1067 and RFC 1098.) Security is based on community strings.
- SNMPv2c—The community string-based Administrative Framework for SNMPv2. SNMPv2c (the "c" is for "community") is an experimental Internet protocol defined in RFC 1901, RFC 1905, and RFC 1906. SNMPv2c is an update of the protocol operations and data types of SNMPv2p (SNMPv2 Classic) and uses the community-based security model of SNMPv1.

SNMPv1 and SNMPv2 are not as secure as SNMPv3. SNMP version 1 and 2 use plain text communities and do not perform the authentication or security checks that SNMP version 3 performs. To configure the SNMP Support over VPNs—Context-Based Access Control feature when using SNMP version 1 or SNMP version 2, you need to associate a community name with a VPN. This association causes SNMP to process requests coming in for a particular community string only if it comes in from the configured VRF. If the community string contained in the incoming packet does not have an associated VRF, it is processed only if it came in through a non-VRF interface. This process prevents users outside the VPN from snooping a clear text community string to query the VPN's data. These methods of source address validation are not as secure as using SNMPv3.

SNMP Notification Support over VPNs

The SNMP Notification Support over VPNs feature allows the sending and receiving of SNMP notifications (traps and informs) using VPN routing and forwarding (VRF) instance tables. In particular, this feature adds support to Cisco software for the sending and receiving of SNMP notifications (traps and informs) specific to individual VPNs.

SNMP is an application-layer protocol that provides a message format for communication between SNMP managers and agents.

A VPN is a network that provides high-connectivity transfers on a shared system with the same usage guidelines as a private network. A VPN can be built on the Internet over IP, Frame Relay, or ATM networks.

A VRF stores per-VPN routing data. It defines the VPN membership of a customer site attached to the network access server (NAS). A VRF consists of an IP routing table, a derived Cisco Express Forwarding (formerly known as CEF) table, and guidelines and routing protocol parameters that control the information that is included in the routing table.

The SNMP Support for VPNs—Context-Based Access Control feature provides configuration commands that allow users to associate SNMP agents and managers with specific VRFs. The associated VRF is used for the sending of SNMP notifications (traps and informs) and responses between agents and managers. If a VRF is not specified, the default routing table for the VPN is used.

VPN-Aware SNMP

The SNMP Support for VPNs—Context-Based Access Control feature extends the capabilities of the SNMP Notification Support for VPNs feature and enables SNMP to differentiate between incoming packets from different VPNs.

When the SNMP Support for VPNs—Context-Based Access Control feature is configured, SNMP accepts requests on any configured VRF and returns responses to the same VRF. A trap host also can be associated with a specific VRF. The configured VRF is then used for sending out traps; otherwise, the default routing table is used. You also can associate a remote user with a specific VRF. You also can configure the VRFs from which SNMP should accept requests. Any requests coming from VRFs that are not specified are dropped.

IP access lists can be configured and associated with SNMP community strings. This feature enables you to configure an association between VRF instances with SNMP community strings. When a VRF instance is associated with an SNMP community string, SNMP processes the requests coming in for a particular community string only if the requests are received from the configured VRF. If the community string contained in the incoming packet does not have a VRF associated with it, the community string will be processed only if it came in through a non-VRF interface.

You also can enable or disable authentication traps for SNMP packets dropped due to VRF mismatches. By default if SNMP authentication traps are enabled, VRF authentication traps are also enabled.

VPN Route Distinguishers

A route distinguisher (RD) creates routing and forwarding tables and specifies the default route distinguisher for a VPN. The RD is added to the beginning of the customer's IPv4 prefixes to change them into globally unique VPN-IPv4 prefixes.

The RD is either an autonomous system number (ASN)-relative RD, in which case it comprises an autonomous system number and an arbitrary number, or it is an IP-address-relative RD, in which case it comprises an IP address and an arbitrary number.

You can enter an RD in either of these formats:

- 16-bit ASN: your 16-bit number: For example, 101:3.
- 32-bit IP address: your 32-bit number: For example, 192.168.122.15:1.

SNMP Contexts

SNMP contexts provide VPN users with a secure way of accessing MIB data. When a VPN is associated with a context, that VPN's specific MIB data exists in that context. Associating a VPN with a context enables service providers to manage networks with multiple VPNs. Creating and associating a context with a VPN enables a provider to prevent the users of one VPN from accessing information about other VPN userss on the same networking device.

VPN-aware SNMP requires an agreement between SNMP manager and agent entities operating in a VPN environment on a mapping between the SNMP security name and the VPN ID. This mapping is created by using multiple contexts for the SNMP data of different VPNs through the configuration of the SNMP-VACM-MIB. The SNMP-VACM-MIB is configured with views so that a user on a VPN with a security name is allowed access to the restricted object space associated with a user's access type in the context associated with the user of that VPN.

SNMP request messages undergo three phases of security and access control before a response message is sent back with the object values in the context of a VPN:

- In the first phase, the username is authenticated. This phase ensures that the user is authenticated and authorized for SNMP access.
- In the second phase, the user is authorized for the SNMP access requested to the group objects under consideration of the configured SNMP context. This phase is called the access control phase.
- In the third phase, access is made to a particular instance of a table entry. With this third phase, complete retrieval can be based on the SNMP context name.

How to Configure SNMP Support over VPNs—Context-Based Access Control

Configuring an SNMP Context and Associating the SNMP Context with a VPN

Perform this task to configure an SNMP context and to associate the SNMP context with a VPN.

Note

- Only the following MIBs are context-aware. All the tables in these MIBs can be polled:
  - CISCO-IPSEC-FLOW-MONITOR-MIB
  - CISCO-IPSEC-MIB
  - CISCO-PING-MIB
  - IP-FORWARD-MIB
  - MPLS-LDP-MIB
- Only two SNMP variables in the IP-FORWARD-MIB can be polled: 1.3.6.1.2.1.4.24.3 (ipCidrRouteNumber - Scalar) and 1.3.6.1.2.1.4.24.4.1 (ipCidrRouteEntry - Table).

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server context** *context-name*
4. **ip vrf** *vrf-name*
5. **rd** *route-distinguisher*
6. **context** *context-name*
7. **route-target** {**import** | **export** | **both**} *route-target-ext-community*
8. **end**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server context** *context-name* **Example:** `Device(config)# snmp-server context context1`	Creates and names an SNMP context.

	Command or Action	Purpose
Step 4	**ip vrf** *vrf-name* **Example:** `Device(config)# ip vrf vrf1`	Configures a VRF routing table and enters VRF configuration mode.
Step 5	**rd** *route-distinguisher* **Example:** `Device(config-vrf)# rd 100:120`	Creates a VPN route distinguisher.
Step 6	**context** *context-name* **Example:** `Device(config-vrf)# context context1`	Associates an SNMP context with a particular VRF. **Note** Depending on your release, the **context** command is replaced by the **snmp context** command. See the *Cisco IOS Network Management Command Reference* for more information.
Step 7	**route-target** {**import** | **export** | **both**} *route-target-ext-community* **Example:** `Device(config-vrf)# route-target export 100:1000`	(Optional) Creates a route-target extended community for a VRF.
Step 8	**end** **Example:** `Device(config-vrf)# end`	Exits interface mode and enters global configuration mode.
Step 9	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Configuring SNMP Support and Associating an SNMP Context

Perform this task to configure SNMP support and associate it with an SNMP context.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server user** *username group-name* [**remote** *host* [**udp-port** *port*] [**vrf** *vrf-name*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**access** [**ipv6** *nacl*] [**priv** {**des** | **3des** | **aes** {**128** | **192** | **256**}} *privpassword*] {*acl-number* | *acl-name*}]
4. **snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**context** *context-name*] [**read** *read-view*] [**write** *write-view*] [**notify** *notify-view*] [**access** [**ipv6** *named-access-list*] [*acl-number*| *acl-name*]]
5. **snmp-server view** *view-name oid-tree* {**included** | **excluded**}
6. **snmp-server enable traps** [*notification-type*] [**vrrp**]
7. **snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**ipv6** *nacl*] [*access-list-number* | *extended-access-list-number* | *access-list-name*]
8. **snmp-server host** {*hostname* | *ip-address*} [**vrf** *vrf-name*] [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*]
9. **snmp mib community-map** *community-name* [**context** *context-name*] [**engineid** *engine-id*] [**security-name** *security-name*][**target-list** *upn-list-name*]
10. **snmp mib target list** *vpn-list-name* {**vrf** *vrf-name* | **host** *ip-address*}
11. **no snmp-server trap authentication vrf**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server user** *username group-name* [**remote** *host* [**udp-port** *port*] [**vrf** *vrf-name*]] {**v1**	**v2c**	**v3** [**encrypted**] [**auth** {**md5**	**sha**} *auth-password*]} [**access** [**ipv6** *nacl*] [**priv** {**des**	**3des**	**aes** {**128**	**192**	**256**}} *privpassword*] {*acl-number*	*acl-name*}] **Example:** `Device(config)# snmp-server user customer1 group1 v1`	Configures a new user to an SNMP group.
Step 4	**snmp-server group** *group-name* {**v1**	**v2c**	**v3** {**auth**	**noauth**	**priv**}} [**context** *context-name*] [**read** *read-view*] [**write** *write-view*] [**notify** *notify-view*] [**access** [**ipv6** *named-access-list*] [*acl-number*	*acl-name*]]	Configures a new SNMP group or a table that maps SNMP users to SNMP views.

	Command or Action	Purpose
	Example: `Device(config)# snmp-server group group1 v1 context context1 read view1 write view1 notify view1`	• Use the **context** *context-name* keyword argument pair to associate the specified SNMP group with a configured SNMP context.
Step 5	**snmp-server view** *view-name oid-tree* {**included** | **excluded**} **Example:** `Device(config)# snmp-server view view1 ipForward included`	Creates or updates a view entry.
Step 6	**snmp-server enable traps** [*notification-type*] [**vrrp**] **Example:** `Device(config)# snmp-server enable traps`	Enables all SNMP notifications (traps or informs) available on your system.
Step 7	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [**ipv6** *nacl*] [*access-list-number* | *extended-access-list-number* | *access-list-name*] **Example:** `Device(config)# snmp-server community public view view1 rw`	Sets up the community access string to permit access to the SNMP.
Step 8	**snmp-server host** {*hostname* | *ip-address*} [**vrf** *vrf-name*] [**traps** | **informs**] [**version** {**1** | **2c** | **3** [**auth** | **noauth** | **priv**]}] *community-string* [**udp-port** *port*] [*notification-type*] **Example:** `Device(config)# snmp-server host 10.0.0.1 vrf vrf1 public udp-port 7002`	Specifies the recipient of an SNMP notification operation.
Step 9	**snmp mib community-map** *community-name* [**context** *context-name*] [**engineid** *engine-id*] [**security-name** *security-name*][**target-list** *upn-list-name*] **Example:** `Device(config)# snmp mib community-map community1 context context1 target-list commAVpn`	Associates an SNMP community with an SNMP context, Engine ID, or security name.
Step 10	**snmp mib target list** *vpn-list-name* {**vrf** *vrf-name* | **host** *ip-address*} **Example:** `Device(config)# snmp mib target list commAVpn vrf vrf1`	Creates a list of target VRFs and hosts to associate with an SNMP community.

	Command or Action	Purpose
Step 11	**no snmp-server trap authentication vrf** **Example:** `Device(config)# no snmp-server trap authentication vrf`	(Optional) Disables all SNMP authentication notifications (traps and informs) generated for packets received on VRF interfaces. • Use this command to disable authentication traps only for those packets on VRF interfaces with incorrect community associations.

Configuration Examples for SNMP Support over VPNs—Context-Based Access Control

Example: Configuring Context-Based Access Control

The following configuration example shows how to configure the SNMP Support over VPNs—Context-Based Access Control feature for SNMPv1 or SNMPv2:

Note Depending on your releases, the **context** command is replaced by the **snmp context** command. See the *Cisco IOS Network Management Command Reference* for more information.

```
snmp-server context A
snmp-server context B
ip vrf Customer_A
 rd 100:110
 context A
 route-target export 100:1000
 route-target import 100:1000
!
ip vrf Customer_B
 rd 100:120
 context B
 route-target export 100:2000
 route-target import 100:2000
!
interface Ethernet3/1
 description Belongs to VPN A
 ip vrf forwarding CustomerA
 ip address 192.168.2.1 255.255.255.0

interface Ethernet3/2
 description Belongs to VPN B
 ip vrf forwarding CustomerB
 ip address 192.168.2.2 255.255.255.0
snmp-server user commA grp1A v1
snmp-server user commA grp2A v2c
snmp-server user commB grp1B v1
snmp-server user commB grp2B v2c
snmp-server group grp1A v1 context A read viewA write viewA notify viewA
snmp-server group grp1B v1 context B read viewB write viewB notify viewB
```

```
snmp-server view viewA ipForward included
snmp-server view viewA ciscoPingMIB included
snmp-server view viewB ipForward included
snmp-server view viewB ciscoPingMIB included
snmp-server enable traps
snmp-server host 192.168.2.3 vrf CustomerA commA udp-port 7002
snmp-server host 192.168.2.4 vrf CustomerB commB udp-port 7002
snmp mib community-map commA context A target-list commAvpn
! Configures source address validation
snmp mib community-map commB context B target-list commBvpn
! Configures source address validation
snmp mib target list commAvpn vrf CustomerA
! Configures a list of VRFs or from which community commA is valid
snmp mib target list commBvpn vrf CustomerB
! Configures a list of VRFs or from which community commB is valid
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Cisco IOS SNMP Support Command Reference	Cisco IOS SNMP Support Command Reference

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
Standard 58	*Structure of Management Information Version 2 (SMIv2) >*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*

Standard/RFC	Title
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2233	The Interface Group MIB using SMIv2
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	Event MIB
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers	To locate and download MIBs for selected platforms, Cisco IOS XE software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

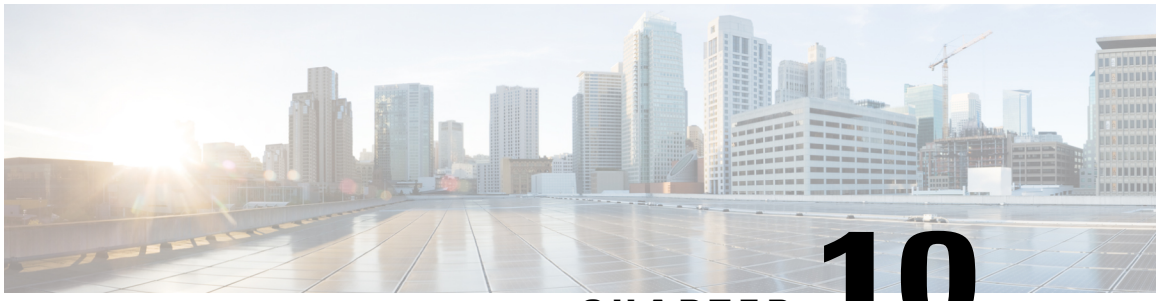
Feature Information for SNMP Support over VPNs—Context-Based Access Control

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 9: Feature Information for SNMP Support over VPNs—Context-Based Access Control

Feature Name	Releases	Feature Information
SNMP Support over VPNs—Context-Based Access Control		The SNMP Support over VPNs—Context-Based Access Control feature provides the infrastructure for multiple SNMP context support in Cisco software and VPN-aware MIB infrastructure using the multiple SNMP context support infrastructure.



CHAPTER 10

Interfaces MIB—SNMP context–based access

The Interfaces MIB—Simple Network Management Protocol (SNMP) context–based access feature provides ability to query the Interfaces MIB objects and the information returned will be restricted to the VRF to which the SNMP context is mapped to. Notification hosts may also be configured with contexts to restrict the notifications that need to be sent to the particular host.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Interfaces MIB—SNMP context–based access

The interface MIB (IF-MIB) has been modified to support context-aware packet information in virtual route forwarding (VRF) environments. VRF environments require that contexts apply to VPNs so that clients can be given selective access to the information stored in the IF-MIB. Clients that belong to a particular VRF can access information about the interface from the IF-MIB that belongs to that VRF only. When a client tries to get information from an interface that is associated with a particular context, the client can see only the information that belongs to that context and cannot see information to which it is not entitled.

No commands have been modified or added to support this feature. This feature is automatically enabled when VRF is configured.

The IF-MIB supports all tables defined in RFC 2863 and the CISCO-IFEXTENSION-MIB.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*

Standard/RFC	Title
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

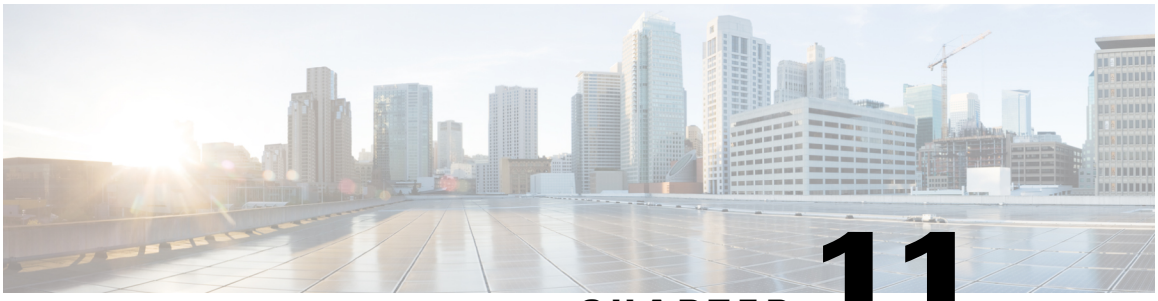
Feature Information for Interfaces MIB—SNMP context–based access

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 10: Feature Information for Interfaces MIB—SNMP context–based access

Feature Name	Releases	Feature Information
Interfaces MIB—SNMP context–based access	12.2(33)SRB 12.2(33)SB 12.2(44)SG 15.0(1)S	TheInterfaces MIB—SNMP context–based access feature provides the ability to query Interfaces MIB objects. The information returned will be restricted to the VRF to which the SNMP context is mapped. Notification hosts may also be configured with contexts to restrict notifications that need to be sent to the particular host.



CHAPTER 11

SNMP Support for VLAN Subinterfaces

This feature module describes the SNMP Support for VLAN Subinterfaces feature. It includes information on the benefits of the new feature, supported platforms, supported standards, and the commands necessary to configure the SNMP Support for VLAN Subinterfaces feature.

The SNMP Support for VLAN Subinterfaces feature provides mib-2 interfaces sparse table support for Fast Ethernet subinterfaces. This enhancement is similar to the functionality supported in Frame Relay subinterfaces.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About SNMP Support for VLAN Subinterfaces

Benefits

Sparse table support for the interfaces table on Fast Ethernet subinterfaces provides customers accustomed to Frame Relay subinterfaces the same functionality.

Supported Platforms

- Cisco 2600 series
- Cisco 3600 series
- Cisco 4000-m series
- Cisco 7200 series
- Cisco 7500 series

How to SNMP Support for VLAN Subinterfaces

Enabling the SNMP Agent on VLAN Subinterfaces

Perform the following task to enable the SNMP agent on VLAN subinterfaces.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp community** *string*
4. **interface** *type slot/port*
5. **encapsulation isl** *vlan-identifier*
6. **ip address** *ip-address mask*
7. **end**
8. **show vlans**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Router> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**snmp community** *string* **Example:** `Router(config)# snmp community public`	Enables the SNMP agent for remote access.
Step 4	**interface** *type slot/port* **Example:** `Router(config)# interface FastEthernet 0/1.1`	Selects a particular Fast Ethernet interface for configuration.
Step 5	**encapsulation isl** *vlan-identifier* **Example:** `Router(config-if)# encapsulation isl 10`	Enables the Inter-Switch Link.
Step 6	**ip address** *ip-address mask* **Example:** `Router(config)# ip address 192.168.10.1 255.255.255.0`	Sets a primary or secondary IP address for an interface.
Step 7	**end** **Example:** `Router(config-if)# end`	Returns to privileged EXEC mode.
Step 8	**show vlans** **Example:** `Router# show vlans`	Displays VLAN subinterfaces.

Configuration Examples for SNMP Support for VLAN Subinterfaces

Example Enabling the SNMP Agent for VLAN Subinterfaces

The following configuration example shows you how to enable the SNMP agent on the router with VLAN subinterfaces to monitor the SNMP application remotely:

```
snmp community public
!
interface FastEthernet4/0.100
 encapsulation isl 100
 ip address 192.168.10.21 255.255.255.0
!
interface FastEthernet4/0.200
 encapsulation isl 200
 ip address 172.21.200.11 255.255.255.0
!
interface FastEthernet4/1.1
 encapsulation isl 10
 ip address 171.69.2.111 255.255.255.0
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
SNMP commands	*Cisco IOS Network Management Command Reference*

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
• None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 1573	*Evolution of the Interfaces Group of MIB-II*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

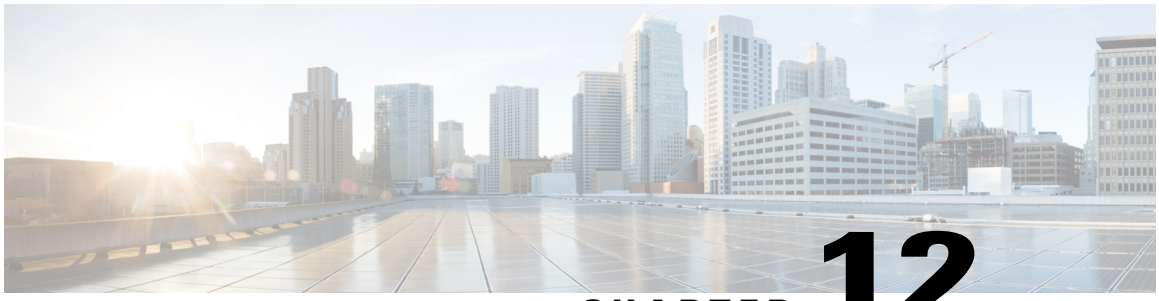
Feature Information for SNMP Support for VLAN Subinterfaces

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 11: Feature Information for SNMP Support for VLAN Subinterfaces

Feature Name	Releases	Feature Information
SNMP Support for VLAN Subinterfaces	12.2	The SNMP Support for VLAN Subinterfaces feature provides mib-2 interfaces sparse table support for Fast Ethernet subinterfaces. This enhancement is similar to the functionality supported in Frame Relay subinterfaces.



CHAPTER 12

Entity MIB—Phase 1

This feature implements the first phase of the Entity MIB, the Logical Entity Table. The Logical Entity Table describes the logical entities managed by a single agent. The Entity MIB also records the time of the last modification to any object in the Entity MIB and sends out a trap when any object is modified. The Entity MIB provides no managed objects with write access.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Entity MIB—Phase 1

Entity MIB—phase 1

The Entity MIB feature implements support for the Entity MIB module, defined in RFC 2037, and provides a mechanism by which a managed device can advertise its logical components, physical components, and logical to physical mappings.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

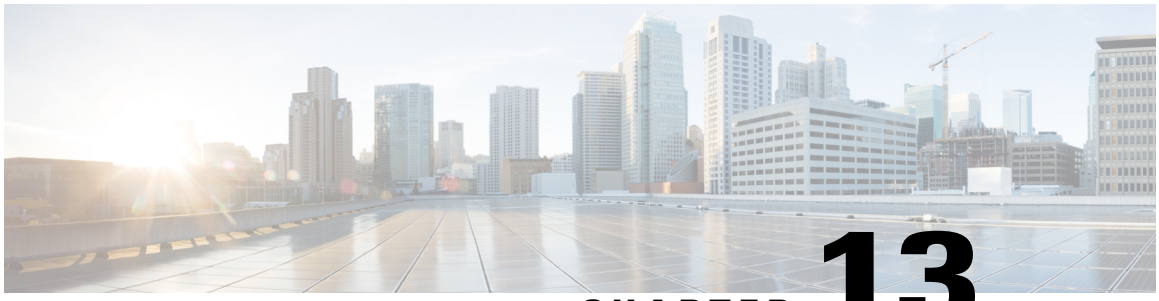
Feature Information for Entity MIB—Phase 1

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 12: Feature Information for Entity MIB - Phase 1

Feature Name	Releases	Feature Information
Entity MIB - Phase 1	11.3(1) 12.0(1) 12.2(2)T 15.0(1)S	The Entity MIB feature implements support for the Entity MIB module, defined in RFC 2037, and provides a mechanism by which a managed device can advertise its logical components, physical components, and logical to physical mappings.



CHAPTER 13

Event MIB and Expression MIB Enhancements

This document provides information about the several existing Simple Network Management Protocol (SNMP) MIBs that are enhanced and new SNMP MIBs that are added.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Event MIB and Expression MIB

Event MIB

The Event MIB provides the ability to monitor MIB objects on a local or remote system using the SNMP, and initiates simple actions whenever a trigger condition is met. For example, an SNMP trap can be generated when an object is modified. When the notifications are triggered through events, the NMS does not need to constantly poll managed devices to track changes.

By allowing the SNMP notifications to take place only when a specified condition is met, Event MIB reduces the load on affected devices and improves the scalability of network management solutions.

The Event MIB operates based on event, object lists configured for the event, event action, trigger, and trigger test.

Events

The event table defines the activities to be performed when an event is triggered. These activities include sending a notification and setting a MIB object. The event table has supplementary tables for additional objects that are configured according to event action. If the event action is set to notification, notifications are sent out whenever the object configured for that event is modified.

Object List

The object table lists objects that can be added to notifications based on trigger, trigger test type, or the event that sends a notification. The Event MIB allows wildcarding, which enables you to monitor multiple instances of an object. To specify a group of object identifiers, you can use the wildcard option.

Trigger

The trigger table defines conditions to trigger events. The trigger table lists the objects to be monitored and associates each trigger with an event. An event occurs when a trigger is activated. To create a trigger, you should configure a trigger entry in the mteTriggerTable of the Event MIB. This trigger entry specifies the object identifier of the object to be monitored. Each trigger is configured to monitor a single object or a group of objects specified by a wildcard (*). The Event MIB process checks the state of the monitored object at specified intervals.

Trigger Test

The trigger table has supplementary tables for additional objects that are configured based on the type of test performed for a trigger. For each trigger entry type such as existence, threshold, or Boolean, the corresponding tables (existence, threshold, and Boolean tables) are populated with the information required to perform the test. The Event MIB allows you to set event triggers based on existence, threshold, and Boolean trigger types. When the specified test on an object returns a value of *true*, the trigger is activated. You can configure the Event MIB to send out notifications to the interested host when a trigger is activated.

Expression MIB

The Expression MIB allows you to create expressions based on a combination of objects. The expressions are evaluated according to the sampling method. The Expression MIB supports the following types of object sampling:

- Absolute
- Delta
- Changed

If there are no delta or change values in an expression, the expression is evaluated when a requester attempts to read the value of the expression. In this case, all requesters get a newly calculated value.

For expressions with delta or change values, an evaluation is performed for every sampling. In this case, requesters get the value as of the last sample period.

Absolute Sampling

Absolute sampling uses the value of the MIB object during sampling.

Delta Sampling

Delta sampling is used for expressions with counters that are identified based on delta (difference) from one sample to the next. Delta sampling requires the application to do continuous sampling, because it uses the value of the last sample.

Changed Sampling

Changed sampling uses the changed value of the object since the last sample.

How to Configure Event MIB and Expression MIB

Configuring Event MIB Using SNMP

The Event MIB can be configured using SNMP directly. In this procedure, the Event MIB is configured to monitor the delta values of ifInOctets for all interfaces once per minute. If any of the samples exceed the specified threshold, a trap notification will be sent.

There are no Cisco software configuration tasks associated with the Event MIB. All configuration of Event MIB functionality must be performed though applications using SNMP. This section provides a sample configuration session using a network management application on an external device. See the "Additional References" section for information about configuring SNMP on your Cisco routing device.

All configuration of Event MIB functionality must be performed though applications using SNMP. The following section provides a step-by-step Event MIB configuration using SNMP research tools available for Sun workstations. The **setany** commands given below are executed using the SNMP application.

Note These are not Cisco command line interface commands. It is assumed that SNMP has been configured on your routing device.

In this configuration, the objective is to monitor ifInOctets for all interfaces. The Event MIB is configured to monitor the delta values of ifInOctets for all interfaces once per minute. If any of the samples exceed the specified threshold of 30, a Trap notification will be sent.

There are five parts to the following example:

Setting the Trigger in the Trigger Table

Perform this task to set the trigger in the trigger table.

SUMMARY STEPS

1. **setany -v2c $ADDRESS private mteTriggerEntryStatus.4.106.111.104.110.1 -i 5**
2. **setany -v2c $ADDRESS private mteTriggerValueID.4.106.111.104.110.1 -d 1.3.6.1.2.1.2.2.1.10**
3. **setany -v2c $ADDRESS private mteTriggerValueIDWildcard.4.106.111.104.110.1 -i 1**
4. **setany -v2c $ADDRESS private mteTriggerTest.4.106.111.104.110.1 -o '20'**
5. **setany -v2c $ADDRESS private mteTriggerFrequency.4.106.111.104.110.1 -g 60**
6. **setany -v2c $ADDRESS private mteTriggerSampleType.4.106.111.104.110.1 -i 2**
7. **setany -v2c $ADDRESS private mteTriggerEnabled.4.106.111.104.110.1 -i 1**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**setany -v2c $ADDRESS private mteTriggerEntryStatus.4.106.111.104.110.1 -i 5**	Creates a trigger row in the table with john as the mteOwner and 1 as the trigger name. • The index is given in decimal representation of the ASCII value of john.1.
Step 2	**setany -v2c $ADDRESS private mteTriggerValueID.4.106.111.104.110.1 -d 1.3.6.1.2.1.2.2.1.10**	Sets the mteTriggerValueID to the OID to be watched. • In this example, the OID to be monitored is ifInOctets.
Step 3	**setany -v2c $ADDRESS private mteTriggerValueIDWildcard.4.106.111.104.110.1 -i 1**	Sets the mteTriggerValueIDWildcard to TRUE to denote a object referenced through wildcarding.
Step 4	**setany -v2c $ADDRESS private mteTriggerTest.4.106.111.104.110.1 -o '20'**	Sets the mteTriggerTest to Threshold.
Step 5	**setany -v2c $ADDRESS private mteTriggerFrequency.4.106.111.104.110.1 -g 60**	Sets the mteTriggerFrequency to 60. This means that ifInOctets are monitored once every 60 seconds.
Step 6	**setany -v2c $ADDRESS private mteTriggerSampleType.4.106.111.104.110.1 -i 2**	Sets the sample type to Delta.
Step 7	**setany -v2c $ADDRESS private mteTriggerEnabled.4.106.111.104.110.1 -i 1**	Enables the trigger.

Creating an Event in the Event Table

Perform this task to create an event in the event table.

SUMMARY STEPS

1. **setany -v2c $ADDRESS private mteEventEntryStatus.4.106.111.104.110.101.118.101.110. 116 -i 5**
2. **setany -v2c $ADDRESS private mteEventEnabled.4.106.111.104.110.101.118.101.110.116 -i 1**
3. **setany -v2c $ADDRESS private mteEventEntryStatus.4.106.111.104.110.101.118.101.110. 116 -i 1**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**setany -v2c $ADDRESS private mteEventEntryStatus.4.106.111.104.110.101.118.101.110. 116 -i 5**	Creates a row in the Event Table. • The mteOwner here is again john, and the event is mteEventName. • The default action is to send out a notification.
Step 2	**setany -v2c $ADDRESS private mteEventEnabled.4.106.111.104.110.101.118.101.110.116 -i 1**	Enables the Event.
Step 3	**setany -v2c $ADDRESS private mteEventEntryStatus.4.106.111.104.110.101.118.101.110. 116 -i 1**	Makes the EventRow active.

Setting and Activating the Trigger Threshold in the Trigger Table

Perform this task to set the trigger threshold in the trigger table.

SUMMARY STEPS

1. **setany -v2c $ADDRESS private mteTriggerThresholdRising.4.106.111.104.110.1 -i 30**
2. **setany -v2c $ADDRESS private mteTriggerThresholdRisingEventOwner.4.106.111.104.110.1 -D "owner"**
3. **setany -v2c $ADDRESS private mteTriggerEntryStatus.4.106.111.104.110.1 -i 1**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**setany -v2c $ADDRESS private mteTriggerThresholdRising.4.106.111.104.110.1 -i 30**	Sets the Rising Threshold value to 30. Note that a row would already exist for john.1 in the Trigger Threshold Table.
Step 2	**setany -v2c $ADDRESS private mteTriggerThresholdRisingEventOwner.4.106.111.104.110.1 -D "owner"**	Points to the entry in the Event Table that specifies the action to be performed.

	Command or Action	Purpose
	Example: `setany -v2c $ADDRESS private mteTriggerThresholdRisingEvent.4.106.111.104.110.1 -D "event"`	
Step 3	**setany -v2c $ADDRESS private mteTriggerEntryStatus.4.106.111.104.110.1 -i 1**	Makes the trigger active.

What to Do Next

To confirm that the above configuration is working, ensure that at least one of the interfaces gets more than 30 packets in a minute. This should cause a trap to be sent out after one minute.

Monitoring and Maintaining Event MIB

Use the following commands to monitor Event MIB activity from the Cisco command line interface:

Command	Purpose
debug management event mib	Prints messages to the screen whenever the Event MIB evaluates a specified trigger. These messages are given in realtime and are intended to be used by technical support engineers for troubleshooting purposes.
show management event	Displays the SNMP Event values that have been configured on your routing device through the use of the Event MIB.

Configuring Event MIB Using Command Line Interface

The Event MIB can be configured using SNMP directly. In this procedure, the Event MIB is configured to monitor delta values of ifInOctets for all interfaces once per minute. If any of the samples exceed the specified threshold, a trap notification will be sent.

Depending on your release, note that the Event MIB feature is enhanced to add command line interface commands to configure the events, event action, and trigger.

This section contains the following tasks to configure the Event MIB:

Configuring Scalar Variables

Perform this task to configure scalar variables for Event MIB.

Before You Begin

To configure the scalar variables for Event MIB, you should be familiar with the Event MIB scalar variables.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib event sample minimum** *value*
4. **snmp mib event sample instance maximum** *value*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib event sample minimum** *value* **Example:** `Device(config)# snmp mib event sample minimum 10`	Sets the minimum value for object sampling.
Step 4	**snmp mib event sample instance maximum** *value* **Example:** `Device(config)# snmp mib event sample instance maximum 50`	Sets the maximum value for object instance sampling.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

Configuring Event MIB Object List

To configure the Event MIB, you need to set up a list of objects that can be added to notifications according to the trigger, trigger test, or event.

Before You Begin

To configure the Event MIB object list, you should be familiar with the Event MIB objects and object identifiers, which can be added to notifications according to the event, trigger, or trigger test.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib event object list owner** *object-list-owner* **name** *object-list-name object-number*
4. **object id** *object-identifier*
5. **wildcard**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib event object list owner** *object-list-owner* **name** *object-list-name object-number* **Example:** `Device(config)# snmp mib event object list owner owner1 name objectA 10`	Configures the Event MIB object list.
Step 4	**object id** *object-identifier* **Example:** `Device(config-event-objlist)# object id ifInOctets`	Specifies the object identifier for the object configured for the event.

	Command or Action	Purpose
Step 5	**wildcard** **Example:** `Device(config-event-objlist)# wildcard`	(Optional) Starts a wildcard search for object identifiers. By specifying a partial object identifier, you can obtain a list of object identifiers.
Step 6	**end** **Example:** `Device(config-event-objlist)# end`	Exits object list configuration mode.

Configuring Event

Perform this task to configure a management event.

Before You Begin

To configure a management event, you should be familiar with SNMP MIB events and object identifiers.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib event owner** *event-owner* **name** *event-name*
4. **description** *event-description*
5. **enable**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**snmp mib event owner** *event-owner* **name** *event-name* **Example:** `Device(config)# snmp mib event owner owner1 event EventA`	Enters event configuration mode.
Step 4	**description** *event-description* **Example:** `Device(config-event)# description "EventA is an RMON event"`	Describes the function and use of the event.
Step 5	**enable** **Example:** `Device(config-event)# enable`	Enables the event. **Note** The event can be executed during an event trigger only if it is enabled.
Step 6	**end** **Example:** `Device(config-event)# end`	Exits event configuration mode.

Configuring Event Action

By configuring an event action, you can define the actions that an application can perform during an event trigger. The actions for an event include sending a notification, setting a MIB object and so on. You can set the event action information to either **set** or **notification**. The actions for the event can be configured only in event configuration mode.

The following sections contain the tasks to configure an event action:

Configuring Action Notification

Perform this task to set the notification action for the event.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib event owner** *event-owner* **name** *event-name*
4. **action notification**
5. **object id** *object-id*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib event owner** *event-owner* **name** *event-name* **Example:** `Device(config)# snmp mib event owner owner1 name test`	Enters event configuration mode.
Step 4	**action notification** **Example:** `Device(config-event)# action notification`	Sets the notification action for an event and enters action notification configuration mode. **Note** If the event action is set to notification, a notification is generated whenever an object associated with an event is modified.
Step 5	**object id** *object-id* **Example:** `Device(config-event-action-notification)# object id ifInOctets`	Configures an object for action notification. When the object specified is modified, a notification will be sent to the host system.
Step 6	**end** **Example:** `Device(config-event-action-notification)# end`	Exits action notification configuration mode.

Configuring Action Set

Perform this task to set actions for an event.

SUMMARY STEPS

1. **action set**
2. **object id** *object-id*
3. **value** *integer-value*
4. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**action set** **Example:** `Device(config-event)# action set`	Enters action set configuration mode.
Step 2	**object id** *object-id* **Example:** `Device(config-event-action-set)# object id ifInOctets`	Configures an object for action set. • When the object specified is modified, a specified action will be performed.
Step 3	**value** *integer-value* **Example:** `Device(config-event-action-set)# value 10`	Sets a value for the object.
Step 4	**end** **Example:** `Device(config-event-action-set)# end`	Exits action set configuration mode.

Configuring Event Trigger

By configuring an event trigger, you can list the objects to monitor, and associate each trigger to an event.

Perform this task to configure an event trigger.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib event trigger owner** *trigger-owner* **name** *trigger-name*
4. **description** *trigger-description*
5. **frequency** *seconds*
6. **object list owner** *object-list-owner* **name** *object-list-name*
7. **object id** *object-identifier*
8. **enable**
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib event trigger owner** *trigger-owner* **name** *trigger-name* **Example:** `Device(config)# snmp mib event trigger owner owner1 name EventTriggerA`	Enables event trigger configuration mode for the specified event trigger.
Step 4	**description** *trigger-description* **Example:** `Device(config-event-trigger)# description "EventTriggerA is an RMON alarm."`	Describes the function and use of the event trigger.
Step 5	**frequency** *seconds* **Example:** `Device(config-event-trigger)# frequency 120`	Configures the waiting time (number of seconds) between trigger samples.
Step 6	**object list owner** *object-list-owner* **name** *object-list-name* **Example:** `Device(config-event-trigger)# object list owner owner1 name ObjectListA`	Specifies the list of objects that can be added to notifications.
Step 7	**object id** *object-identifier* **Example:** `Device(config-event-trigger)# object id ifInOctets`	Configures object identifiers for an event trigger.
Step 8	**enable** **Example:** `Device(config-event-trigger)# enable`	Enables the event trigger.

	Command or Action	Purpose
Step 9	**end** **Example:** `Device(config-event-trigger)# end`	Exits event trigger configuration mode.

Configuring Existence Trigger Test

You should configure this trigger type in event trigger configuration mode.

Perform this task to configure trigger parameters for the test existence trigger type.

SUMMARY STEPS

1. **test existence**
2. **event owner** *event-owner* **name** *event-name*
3. **object list owner** *object-list-owner* **name** *object-list-name*
4. **type** {**present** | **absent** | **changed**}
5. **startup** {**present** | **absent**}
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**test existence** **Example:** `Device(config-event-trigger)# test existence`	Enables test existence configuration mode.
Step 2	**event owner** *event-owner* **name** *event-name* **Example:** `Device(config-event-trigger-existence)# event owner owner1 name EventA`	Configures the event for the existence trigger test.
Step 3	**object list owner** *object-list-owner* **name** *object-list-name* **Example:** `Device(config-event-trigger-existence)# object list owner owner1 name ObjectListA`	Configures the list of objects for the existence trigger test.
Step 4	**type** {**present**	**absent**	**changed**}	Performs the specified type of existence test.

	Command or Action	Purpose
	Example: `Device(config-event-trigger-existence)# type present`	Existence tests are of the following three types: • Present—Setting type to present tests if the objects that appear during the event trigger exist. • Absent—Setting type to absent tests if the objects that disappear during the event trigger exist. • Changed—Setting type to changed tests if the objects that changed during the event trigger exist.
Step 5	**startup** {**present** | **absent**} **Example:** `Device(config-event-trigger-existence)# startup present`	Triggers an event if the test is performed successfully.
Step 6	**end** **Example:** `Device(config-event-trigger-existence)# end`	Exits existence trigger test configuration mode.

Configuring Boolean Trigger Test

You should configure this trigger test in event trigger configuration mode.

Perform this task to configure trigger parameters for the Boolean trigger type.

SUMMARY STEPS

1. **test boolean**
2. **comparison** {**unequal** | **equal** | **less** | **lessOrEqual** | **greater** | **greaterOrEqual**}
3. **value** *integer-value*
4. **object list owner** *object-list-owner* **name** *object-list-name*
5. **event owner** *event-owner* **name** *event-name*
6. **startup**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**test boolean** **Example:** `Device(config-event-trigger)# test boolean`	Enables Boolean trigger test configuration mode.
Step 2	**comparison** {**unequal** | **equal** | **less** | **lessOrEqual** | **greater** | **greaterOrEqual**} **Example:** `Device(config-event-trigger-boolean)# comparison unequal`	Performs the specified Boolean comparison test. • The value for the Boolean comparison test can be set to unequal, equal, less, lessOrEqual, greater, or greaterOrEqual.
Step 3	**value** *integer-value* **Example:** `Device(config-event-trigger-boolean)# value 10`	Sets a value for the Boolean trigger test.
Step 4	**object list owner** *object-list-owner* **name** *object-list-name* **Example:** `Device(config-event-trigger-boolean)# object list owner owner1 name ObjectListA`	Configures the list of objects for the Boolean trigger test.
Step 5	**event owner** *event-owner* **name** *event-name* **Example:** `Device(config-event-trigger-boolean)# event owner owner1 name EventA`	Configures the event for the Boolean trigger type.
Step 6	**startup** **Example:** `Device(config-event-trigger-boolean)# startup`	Triggers an event if the test is performed successfully.
Step 7	**end** **Example:** `Device(config-event-trigger-boolean)# end`	Exits Boolean trigger test configuration mode.

Configuring Threshold Trigger Test

You should configure this trigger test in event trigger configuration mode.

Perform this task to configure trigger parameters for the threshold trigger test.

SUMMARY STEPS

1. **test threshold**
2. **object list owner** *object-list-owner* **name** *object-list-name*
3. **rising** *integer-value*
4. **rising event owner** *event-owner* **name** *event-name*
5. **falling** *integer-value*
6. **falling event owner** *event-owner* **name** *event-name*
7. **delta rising** *integer-value*
8. **delta rising event owner** *event-owner* **name** *event-name*
9. **delta falling** *integer-value*
10. **delta falling event owner** *event-owner* **name** *event-name*
11. **startup** {**rising** | **falling** | **rising-or-falling**}
12. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**test threshold** **Example:** `Device(config-event-trigger)# test threshold`	Enables threshold trigger test configuration mode.
Step 2	**object list owner** *object-list-owner* **name** *object-list-name* **Example:** `Device(config-event-trigger-threshold)# object list owner owner1 name ObjectListA`	Configures the list of objects for the threshold trigger test.
Step 3	**rising** *integer-value* **Example:** `Device(config-event-trigger-threshold)# rising 100`	Sets the rising threshold to the specified value.
Step 4	**rising event owner** *event-owner* **name** *event-name* **Example:** `Device(config-event-trigger-threshold)# rising event owner owner1 name EventA`	Configures an event for the threshold trigger test for the rising threshold.
Step 5	**falling** *integer-value* **Example:** `Device(config-event-trigger-threshold)# falling 50`	Sets the falling threshold to the specified value.

	Command or Action	Purpose
Step 6	**falling event owner** *event-owner* **name** *event-name* **Example:** `Device(config-event-trigger-threshold)# falling event owner owner1 name EventB`	Configures an event for the threshold trigger test for the falling threshold.
Step 7	**delta rising** *integer-value* **Example:** `Device(config-event-trigger-threshold)# delta rising 30`	Sets the delta rising threshold to the specified value when the sampling method specified for the event trigger is delta.
Step 8	**delta rising event owner** *event-owner* **name** *event-name* **Example:** `Device(config-event-trigger-threshold)# delta rising event owner owner1 name EventC`	Configures an event for the threshold trigger test for the delta rising threshold.
Step 9	**delta falling** *integer-value* **Example:** `Device(config-event-trigger-threshold)# delta falling 10`	Sets the delta falling threshold to the specified value when the sampling method specified for the event trigger is delta.
Step 10	**delta falling event owner** *event-owner* **name** *event-name* **Example:** `Device(config-event-trigger-threshold)# delta falling event owner owner1 name EventAA`	Configures an event for the threshold target test for the delta falling threshold.
Step 11	**startup** {**rising** | **falling** | **rising-or-falling**} **Example:** `Device(config-event-trigger-threshold)# startup rising`	Triggers an event when the threshold trigger test conditions are met.
Step 12	**end** **Example:** `Device(config-event-trigger-threshold)# end`	Exits threshold trigger test configuration mode.

Configuring Expression MIB Using SNMP

Expression MIB can be configured using SNMP directly.

There are no Cisco software configuration tasks associated with Expression MIB. All configurations of the Expression MIB functionality must be performed though applications using SNMP. This section provides a

sample configuration session using a network management application on an external device. See the Additional References section for information about configuring SNMP on your Cisco routing device.

The following section provides a step-by-step Expression MIB configuration using SNMP research tools available for Sun workstations. The **setany** commands given below are executed using the SNMP application. Note that these commands are not Cisco command line interface commands. It is assumed that SNMP has been configured on your routing device.

In the following configuration, a wildcarded expression involving the addition of the counters ifInOctects and ifOutOctects are evaluated.

SUMMARY STEPS

1. **setany -v2c $SNMP_HOST private expResourceDeltaMinimum.0 -i 60**
2. **setany -v2c $SNMP_HOST private expExpressionIndex.116.101.115.116 -g 9**
3. **setany -v2c $SNMP_HOST private expNameStatus.116.101.115.116 -i 5**
4. **setany -v2c $SNMP_HOST private expExpressionComment.9 -D "test expression"**
5. **setany -v2c $SNMP_HOST private expExpression.9 -D '$1 + $2'**
6. **setany -v2c $SNMP_HOST private expObjectID.9.1 -d ifInOctets**
7. **setany -v2c $SNMP_HOST private expObjectSampleType.9.1 -i 2**
8. **setany -v2c $SNMP_HOST private expObjectIDWildcard.9.1 -i 1**
9. **setany -v2c $SNMP_HOST private expObjectStatus.9.1 -i 1**
10. **setany -v2c $SNMP_HOST private expNameStatus.116.101.115.116 -i 1**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**setany -v2c $SNMP_HOST private expResourceDeltaMinimum.0 -i 60**	Sets the minimum delta interval that the system will accept.
Step 2	**setany -v2c $SNMP_HOST private expExpressionIndex.116.101.115.116 -g 9**	Sets the identification number used for identifying the expression. • For example, expName can be 'test', which is ASCII 116.101.115.116.
Step 3	**setany -v2c $SNMP_HOST private expNameStatus.116.101.115.116 -i 5**	Creates an entry in the expNameStatusTable. **Note** When an entry is created in the expNameTable, it automatically creates an entry in the expExpressionTable.
Step 4	**setany -v2c $SNMP_HOST private expExpressionComment.9 -D "test expression"**	Sets the object to a comment to explain the use or meaning of the expression. • Here, the comment is "test expression".
Step 5	**setany -v2c $SNMP_HOST private expExpression.9 -D '$1 + $2'**	Sets the object expExpression to an expression that needs to be evaluated. • In this expression, "$1" corresponds to "ifInOctets", "$2" corresponds to "ifOutOctets", and the expression signifies the addition of the two counter objects.

	Command or Action	Purpose
Step 6	**setany -v2c $SNMP_HOST private expObjectID.9.1 -d ifInOctets** **Example:** `setany -v2c $SNMP_HOST private expObjectID.9.2 -d ifOutOctets`	Specifies the object identifiers used in the expression mentioned in the above set for calculation. • Here, the number "9", suffixed to the object expObjectID, corresponds to the unique identifier used for identifying the expression, and the number "1" following "9" is another unique identifier used for identifying an object within the expression. Set the expObjectID to the two objects used in forming the expression.
Step 7	**setany -v2c $SNMP_HOST private expObjectSampleType.9.1 -i 2** **Example:** `setany -v2c $SNMP_HOST private expObjectSampleType.9.2 -i 2`	Sets the type of sampling to be done for objects in the expression. • There are two types of sampling: a) Absolute b) Delta. Here, the sample type has been set to "Delta".
Step 8	**setany -v2c $SNMP_HOST private expObjectIDWildcard.9.1 -i 1** **Example:** `setany -v2c $SNMP_HOST private expObjectIDWildcard.9.2 -i 1`	Specifies whether the expObjectID is wildcarded or not. In this case, both the expObjectID are wildcarded.
Step 9	**setany -v2c $SNMP_HOST private expObjectStatus.9.1 -i 1** **Example:** `setany -v2c $SNMP_HOST private expObjectStatus.9.2 -i 1`	Sets the rows in the expObjectTable to active.
Step 10	**setany -v2c $SNMP_HOST private expNameStatus.116.101.115.116 -i 1**	Sets the rows in the expNameTable to active so that the value of the expression can be evaluated. • The value of the expression can now be obtained from the expValueTable.

Configuring Expression MIB Using Command Line Interface

Expression MIB can be configured using SNMP directly. Depending on your release, you can find that the Expression MIB feature is enhanced to add command line interface commands to configure expressions. You should be familiar with expressions, object identifiers, and sampling methods before configuring Expression MIB.

The following sections contain the tasks to configure Expression MIB:

Configuring Expression MIB Scalar Objects

Expression MIB has the following scalar objects:

- expResourceDeltaMinimum
- expResourceDeltaWildcardInstanceMaximum

Perform this task to configure Expression MIB scalar objects.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib expression delta minimum** *seconds*
4. **snmp mib expression delta wildcard maximum** *number-of-instances*
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib expression delta minimum** *seconds* **Example:** `Device(config)# snmp mib expression delta minimum 20`	(Optional) Sets the minimum delta interval in seconds. **Note** Application may use larger values for this minimum delta interval to lower the impact of constantly computing deltas. For larger delta sampling intervals, the application samples less often and has less overhead. By using this command, you can enforce a lower overhead for all expressions created after the delta interval is set.
Step 4	**snmp mib expression delta wildcard maximum** *number-of-instances* **Example:** `Device(config)# snmp mib expression delta maximum 120`	(Optional) Limits the maximum number of dynamic instance entries for wildcard delta objects in expressions. • For a given delta expression, the number of dynamic instances is the number of values that meet all criteria to exist, times the number of delta values in the expression. • There is no preset limit for the instance entries and it is dynamic based on a system's resources.

	Command or Action	Purpose
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Configuring Expressions

Perform this task to configure an expression.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib expression owner** *expression-owner* **name** *expression-name*
4. **description** *expression-description*
5. **expression** *expression*
6. **delta interval** *seconds*
7. **value type** {**counter32** | **unsigned32** | **timeticks** | **integer32** | **ipaddress** | **octetstring** | **objectid** | **counter64**}
8. **enable**
9. **object** *object-number*
10. **id** *object-identifier*
11. **wildcard**
12. **discontinuity object** *discontinuity-object-id* [**wildcard**] [**type** {**timeticks** | **timestamp** | **date-and-time**}]
13. **conditional object** *conditional-object-id* **[wildcard]**
14. **sample** {**absolute** | **delta** | **changed**}
15. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**snmp mib expression owner** *expression-owner* **name** *expression-name* **Example:** `Device(config-expression)# snmp mib expression owner owner1 name ExpA`	Enables the expression to be configured.
Step 4	**description** *expression-description* **Example:** `Device(config-expression)# description this expression is created for the sysLocation MIB object`	Configures a description for the expression.
Step 5	**expression** *expression* **Example:** `Device(config-expression)# expression ($1+$2)*800/$3`	Configures the expression to be evaluated. **Note** The expressions are in ANSI C syntax. However, the variables in an expression are defined as a combination of the dollar sign ($) and an integer that corresponds to the object number of the object used in evaluating the expression.
Step 6	**delta interval** *seconds* **Example:** `Device(config-expression)# delta interval 180`	Configures the sampling interval for objects in the expression if the sampling method is delta.
Step 7	**value type** {**counter32** | **unsigned32** | **timeticks** | **integer32** | **ipaddress** | **octetstring** | **objectid** | **counter64**} **Example:** `Device(config-expression)# value type counter32`	Sets the specified value type for the expression.
Step 8	**enable** **Example:** `Device(config-expression)# enable`	Enables an expression for evaluation.
Step 9	**object** *object-number* **Example:** `Device(config-expression)# object 2`	Configures the objects that are used for evaluating an expression. • The object number is used to associate the object with the variables in the expression. The variable corresponding to the object is $ and object number. Thus, the variable in the example used here corresponds to $10.
Step 10	**id** *object-identifier* **Example:** `Device(config-expression-object)# id ifInOctets`	Configures the object identifier.

	Command or Action	Purpose
Step 11	**wildcard** **Example:** `Device(config-expression-object)# wildcard`	(Optional) Enables a wildcarded search for objects used in evaluating an expression.
Step 12	**discontinuity object** *discontinuity-object-id* [**wildcard**] [**type** {**timeticks** | **timestamp** | **date-and-time**}] **Example:** `Device(config-expression-object)# discontinuity object sysUpTime`	(Optional) Configures the discontinuity properties for the object if the object sampling type is set to delta or changed. The discontinuity object ID supports normal checking for a discontinuity in a counter. • Using the **wildcard** keyword, you can enable wildcarded search for objects with discontinuity properties. • Using the **type** keyword, you can set value for objects with discontinuity properties.
Step 13	**conditional object** *conditional-object-id* **[wildcard]** **Example:** `Device(config-expression-object)# conditional object mib-2.90.1.3.1.1.2.3.112.99.110.4.101.120.112.53`	(Optional) Configures the conditional object identifier. • Using the **wildcard** keyword, you can enable a wildcarded search for conditional objects with discontinuity properties.
Step 14	**sample** {**absolute** | **delta** | **changed**} **Example:** `Device(config-expression-object)# sample delta`	Enables the specified sampling method for the object. This example uses the delta sampling method. You can set any of the three sampling methods: absolute, delta, and changed. • Absolute sampling—Uses the value of the MIB object during sampling. • Delta sampling—Uses the last sampling value maintained in the application. This method requires applications to do continuous sampling. • Changed sampling—Uses the changed value of the object since the last sample.
Step 15	**end** **Example:** `Device(config-expression-object)# end`	Exits expression object configuration mode.

Configuration Examples for Event MIB and Expression MIB

Example: Configuring Event MIB from SNMP

The following example shows how to configure scalar variables for an event:

```
Device# configure terminal
Device(config)# snmp mib event sample minimum 10
Device(config)# snmp mib event sample instance maximum 50
Device(config)# end
```

The following example shows how to configure object list for an event:

```
Device# configure terminal
Device(config)# snmp mib event object list owner owner1 name objectA number 1
Device(config-event-objlist)# object id ifInOctets
Device(config-event-objlist)# wildcard
Device(config-event-objlist)# end
```

The following example shows how to configure an event:

```
Device# configure terminal
Device(config)# snmp mib event owner owner1 event EventA
Device(config-event)# description "eventA is an RMON event."
Device(config-event)# enable
Device(config-event)# end
```

The following example shows how to set the notification action for an event:

```
Device(config-event)# action notification
Device(config-event-action-notification)# object id ifInOctets
Device(config-event-action-notification)# end
```

The following example shows how to set actions for an event:

```
Device(config-event)# action set
Device(config-event-action-set)# object id ifInOctets
Device(config-event-action-set)# value 10
Device(config-event-action-set)# end
```

The following example shows how to configure trigger for an event:

```
Device# configure terminal
Device(config)# snmp mib event trigger owner owner1 name EventTriggerA
Device(config-event-trigger)# description EventTriggerA is an RMON alarm.
Device(config-event-trigger)# frequency 120
Device(config-event-trigger)# object list owner owner1 name ObjectListA
Device(config-event-trigger)# object id ifInOctets
Device(config-event-trigger)# enable
Device(config-event-trigger)# end
```

The following example shows how to configure existence trigger test:

```
Device(config-event-trigger)# test existence
Device(config-event-trigger-existence)# event owner owner1 name EventA
Device(config-event-trigger-existence)# object list owner owner1 name ObjectListA
Device(config-event-trigger-existence)# type present
Device(config-event-trigger-existence)# startup present
Device(config-event-trigger-existence)# end
```

The following example shows how to configure Boolean trigger test:

```
Device(config-event-trigger)# test boolean
Device(config-event-trigger-boolean)# comparison unequal
```

```
Device(config-event-trigger-boolean)# value 10
Device(config-event-trigger-boolean)# object list owner owner1 name ObjectListA
Device(config-event-trigger-boolean)# event owner owner1 name EventA
Device(config-event-trigger-boolean)# startup
Device(config-event-trigger-boolean)# end
```

The following example shows how to configure threshold trigger test:

```
Device(config-event-trigger)# test threshold
Device(config-event-trigger-threshold)# object list owner owner1 name ObjectListA
Device(config-event-trigger-threshold)# rising 100
Device(config-event-trigger-threshold)# rising event owner owner1 name EventA
Device(config-event-trigger-threshold)# falling 50
Device(config-event-trigger-threshold)# falling event owner owner1 name EventA
Device(config-event-trigger-threshold)# delta rising 30
Device(config-event-trigger-threshold)# delta rising event owner owner1 name EventA
Device(config-event-trigger-threshold)# delta falling 10
Device(config-event-trigger-threshold)# delta falling event owner owner1 name EventA
Device(config-event-trigger-threshold)# startup rising
Device(config-event-trigger-threshold)# end
```

Example: Configuring Expression MIB from SNMP

The following example shows how to configure the Expression MIB by using the **snmp mib expression** command in global configuration mode:

```
Device(config)# snmp mib expression owner pcn name exp6
Device(config-expression)# description this expression is created for the sysLocation MIB
object
Device(config-expression)# expression ($1+$2)*800/$3
Device(config-expression)# delta interval 120
Device(config-expression)# value type counter32
Device(config-expression)# enable
Device(config-expression)# object 2
Device(config-expression-object)# id ifInOctets
Device(config-expression-object)# wilcard
Device(config-expression-object)# discontinuity object sysUpTime
Device(config-expression-object)# conditional object
mib-2.90.1.3.1.1.2.3.112.99.110.4.101.120.112.53 wildcard
Device(config-expression-object)# sample delta
Device(config-expression-object)# end
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module

Related Topic	Document Title
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*

Standard/RFC	Title
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

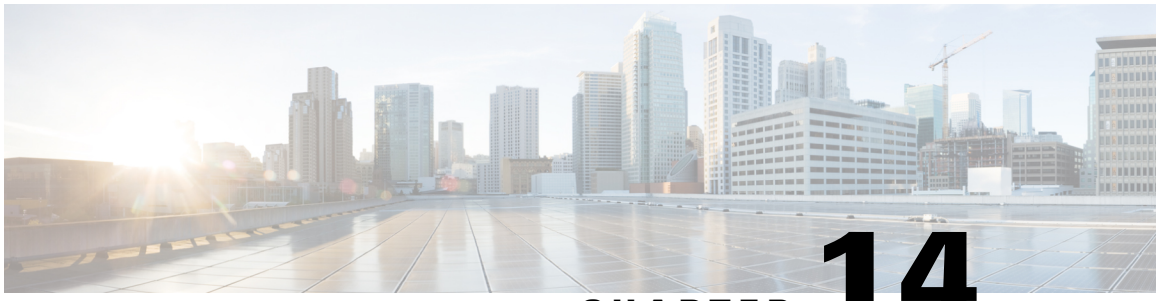
Feature Information for Event MIB and Expression MIB Enhancements

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 13: Feature Information for Event MIB and Expression MIB Enhancements

Feature Name	Releases	Feature Information
Event MIB and Expression MIB Enhancements	12.2(33)SRE 12.2(50)SY 12.4(20)T 15.0(1)S	The Event MIB and Expression MIB feature introduces command line interface commands to configure the Event MIB and Expression MIB. The following commands were introduced or modified: **action (event)**, **comparison**, **conditional object**, **delta (test threshold)**, **delta interval**, **description (event)**, **description (expression)**, **description (trigger)**, **discontinuity object (expression)**, **enable (event)**, **enable (expression)**, **event owner**, **expression**, **falling (test threshold)**, **frequency (event trigger)**, **object (expression)**, **object id**, **object list**, **rising (test threshold)**, **sample (expression)**, **snmp mib event object list**, **snmp mib event owner**, **snmp mib event trigger owner**, **snmp mib expression delta**, **snmp mib expression owner**, **startup (test boolean)**, **startup (test existence)**, **startup (test threshold)**, **test (event trigger)**, **type (test existence)**, **value (test boolean)**, **value type**, and **wildcard (expression)**.



CHAPTER 14

Expression MIB Support of Delta, Wildcarding, and Aggregation

The Expression MIB Support of Delta, Wildcarding, and Aggregation feature adds support of Delta, Wildcarding, and Aggregation to the Expression MIB implementation.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Expression MIB Support of Delta, Wildcarding, and Aggregation

Expression MIB Support of Delta, Wildcarding, and Aggregation

Expression MIB adds support of the Delta, Wildcarding, and Aggregation features in the Distributed Management Expression MIB (EXPRESSION-MIB) to Cisco software for use by SNMP.

The Delta function enables the Expression MIB to use Delta values of an object instead of absolute values when evaluating an expression. Delta is obtained by taking the difference between the current value of an object and its previous value.

The Wildcarding function of the Expression MIB allows evaluation of multiple instances of an object. This is useful in cases where an expression needs to be applied to all instances of an object. The user need not individually specify all instances of an object in the Expression but only has to set the "expWildcardedObject" in "expObjectTable" to TRUE for the respective object.

Aggregation is performed using the sum function in the Expression MIB. The operand to the sum function has to be a wildcard object. The result of the sum function is the sum of values of all instances of the wildcard object.

For a complete description of Expression MIB functionality, see the Distributed Management Expression MIB, Internet-Draft, available through the ITEF at http://tools.ietf.org/html/draft-ietf-disman-express-mib-11.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

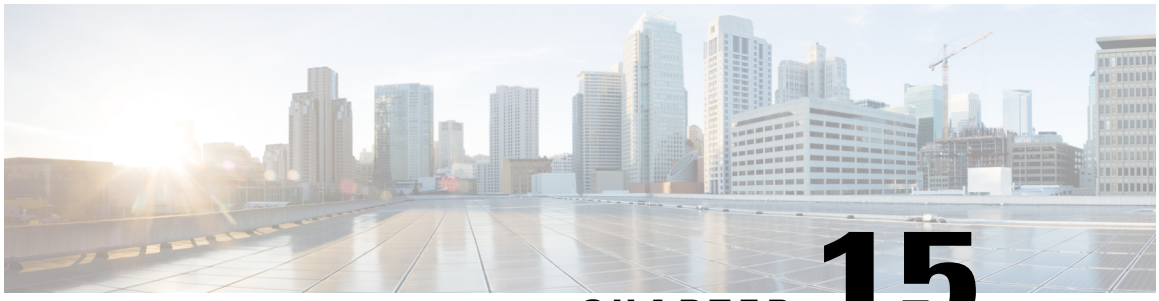
Feature Information for Expression MIB Support of Delta, Wildcarding, and Aggregation

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 14: Feature Information for Expression MIB Support of Delta, Wildcarding, and Aggregation

Feature Name	Releases	Feature Information
Expression MIB Support of Delta, Wildcarding, and Aggregation	12.1(3)T 15.0(1)S	The Expression MIB Support of Delta, Wildcarding, and Aggregation feature adds support of Delta, Wildcarding, and Aggregation to the Expression MIB implementation.



CHAPTER 15

MIB Persistence

The MIB Persistence feature allows the Simple Network Management Protocol (SNMP) data of a MIB to be persistent across reloads; that is, the MIB information retains the same set object values each time a networking device reboots.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about MIB Persistence

MIB Persistence

The MIB Persistence feature allows the SNMP data of a MIB to be persistent across reloads; that is, the MIB information retains the same set object values each time a networking device reboots. MIB Persistence is enabled by issuing the **snmp mib persist** command, and the MIB data of all MIBs that have had persistence enabled using this command is then written to NVRAM by issuing the **write mib-data** command. All modified MIB data must be written to NVRAM using the **write mib-data** command.

Both Event and Expression MIBs allow you to configure a value for an object and to set up object definitions. Both allow rows of data to be modified while the row is in an active state.

Scalar objects are stored every time they are changed, and table entries are stored only if the row is in an active state. The Event MIB has two scalar objects and nine tables to be persisted into NVRAM. The tables are as follows:

- mteEventNotificationTable
- mteEventSetTable
- mteEventTable
- mteObjectsTable
- mteTriggerBooleanTable
- mteTriggerDeltaTable
- mteTriggerExistenceTable
- mteTriggerTable
- mteTriggerThresholdTable

The Expression MIB has two scalar objects and three tables to be stored in NVRAM. The scalar objects are expResourceDeltaMinimum and expResourceDeltaWildcardInstanceMaximum. The tables are as follows:

- expExpressionTable
- expNameTable
- expObjectTable

Writing MIB data to NVRAM may take several seconds. The length of time depends on the amount of MIB data.

Event MIB Persistence and Expression MIB Persistence both allow MIB objects to be saved from reboot to reboot, allowing long-term monitoring of specific devices and interfaces, and configurations of object values that are preserved across reboots.

How to Configure MIB Persistence

Configuring MIB Persistence

Note Depending on your release, configuration of MIB persistence is automatic and is not required to perform manual configuration.

The MIB Persistence features allow the SNMP data of a MIB to be persistent across reloads, that is, MIB information retains the same set of object values each time a networking device reboots. The following sections contain tasks for using Distributed Management Event and Expression MIB persistence.

Prerequisites

- SNMP is configured on your networking device.
- Values for Event MIB and Expression MIB have been configured.

Restrictions

- If the number of MIB objects to persist increases, the NVRAM storage capacity may be strained. Occasionally, the time taken to write MIB data to NVRAM may be longer than expected.
- The Distributed Management Event MIB Persistence feature is not supported on all Cisco platforms. Use Cisco Feature Navigator to find information about platform support and Cisco software image support.

Enabling and Disabling Event MIB Persistence

Perform this task to configure Event MIB Persistence.

Note Event MIB Persistence is disabled by default.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp mib persist event**
4. **no snmp mib persist event**
5. **end**
6. **write mib-data**
7. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 3	**snmp mib persist event** **Example:** `Device(config)# snmp mib persist event`	Enables MIB Persistence for the Event MIB.
Step 4	**no snmp mib persist event** **Example:** `Device(config)# no snmp mib persist event`	(Optional) Disables MIB Persistence for the Event MIB.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 6	**write mib-data** **Example:** `Device# write mib-data`	Saves the Event MIB Persistence configuration data to NVRAM.
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Enabling and Disabling Expression MIB Persistence

Perform this task to configure Expression MIB Persistence.

Note Expression MIB Persistence is disabled by default.

SUMMARY STEPS

1. enable
2. configure terminal
3. snmp mib persist expression
4. no snmp mib persist expression
5. end
6. write mib-data
7. copy running-config startup-config
8. more system:running-config

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp mib persist expression** **Example:** `Device(config)# snmp mib persist expression`	Enables MIB Persistence for Expression MIB.
Step 4	**no snmp mib persist expression** **Example:** `Device(config)# no snmp mib persist expression`	(Optional) Disables MIB Persistence for Expression MIB.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 6	**write mib-data** **Example:** `Device# write mib-data`	Saves the Expression MIB Persistence configuration data to NVRAM.

	Command or Action	Purpose
Step 7	**copy running-config startup-config** **Example:** `Device# copy running-config startup-config`	Copies the running configuration to the startup configuration.
Step 8	**more system:running-config** **Example:** `Device# more system:running-config`	Displays the currently running configuration. • Use this command to verify the MIB persistence configuration.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*

Standard/RFC	Title
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

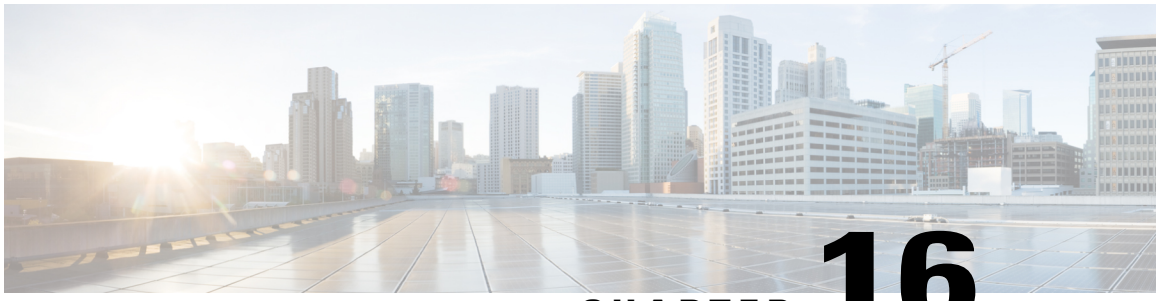
Feature Information for MIB Persistence

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 15: Feature Information for MIB Persistence

Feature Name	Releases	Feature Information
MIB Persistence	12.0(5)T 12.0(12)S 12.1(3)T 12.2(4)T 12.2(4)T3	The MIB Persistence feature allows the SNMP data of a MIB to be persistent across reloads; this means MIB information retains the same set object values each time a networking device reboots. MIB Persistence is enabled by using the **snmp mib persist** command, and the MIB data of all MIBs that have had persistence enabled using this command is then written to NVRAM storage by using the **write mib-data** command. Any modified MIB data must be written to the NVRAM memory using the **write mib-data** command.



CHAPTER 16

Circuit Interface Identification Persistence for SNMP

The Circuit Interface Identification Persistence for SNMP feature maintains the user-defined name of the circuit (defined in the cciDescr object) across reboots, and allows the advanced users of Simple Network Management Protocol (SNMP) to consistently identify the circuits.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Circuit Interface Identification Persistence for SNMP

Circuit Interface Identification Persistence

The Circuit Interface MIB (CISCO-CIRCUIT-INTERFACE-MIB) provides a MIB object (cciDescr) that can be used to identify individual circuit-based interfaces for SNMP monitoring. The Circuit Interface Identification Persistence for SNMP feature maintains this user-defined name of the circuit across reboots, allowing the consistent identification of circuit interfaces. Circuit Interface Identification Persistence is enabled using the **snmp mib persist circuit** global configuration command.

The Circuit Interface Identification Persistence for SNMP feature was introduced with the CSCds67851. The Circuit Interface MIB (CISCO-CIRCUIT-INTERFACE-MIB) provides a MIB object (cciDescr) that can be used to identify individual circuit-based interfaces for SNMP monitoring. The Cisco Circuit Interface Identification MIB was introduced in CSCdp81924.

The Circuit Interface Identification Persistence for SNMP feature maintains the user-defined name of the circuit (defined in the cciDescr object) across reboots, allowing for the consistent identification of circuits.

The Circuit Interface Identification Persistence for SNMP feature is a supplement to the Interface Index Persistence feature. Circuit Interface Identification Persistence is enabled using the **snmp mib persist circuit** global configuration command. Use this command if you need to consistently identify circuits using SNMP across reboots. This command is disabled by default because this feature uses NVRAM.

In addition, the **show snmp mib ifmib ifindex** EXEC mode command allows you to display the Interfaces MIB ifIndex values directly on your system without an NMS; the **show snmp mib** EXEC mode command allows you to display a list of MIB module identifiers registered directly on your system with an NMS. The **snmp ifmib ifalias long** command allows you to specify a description for interfaces or subinterfaces of up to 256 characters in length. Prior to the introduction of this command, ifAlias descriptions for SNMP management were limited to 64 characters.

How to Configure Circuit Interface Identification Persistence for SNMP

Configuring Interface Index Display and Interface Indexes and Long Name Support

The display of Interface Indexes lets advanced users of SNMP view information about the interface registrations directly on a managed agent. An external NMS is not required.

Configuration of Long Alias Names for the interfaces lets users configure the ifAlias (the object defined in the MIB whose length is restricted to 64) up to 255 bytes.

Before You Begin

SNMP must be enabled on your system.

The Interface Index Display and Interface Alias Long Name Support feature is not supported on all Cisco platforms. Use Cisco Feature Navigator to find information about platform support and software image support.

Perform this task to configure the IF-MIB to retain ifAlias values of longer than 64 characters and to configure the ifAlias values for an interface.

Note To verify if the ifAlias description is longer than 64 characters, perform an SNMP MIB walk for the ifMIB ifAlias variable from an NMS and verify that the entire description is displayed in the values for ifXEntry.18. The description for interfaces also appears in the output from the **more system:running config** privileged EXEC mode command.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp ifmib ifalias long**
4. **interface** *type number*
5. **description** *text-string*
6. **end**
7. **show snmp mib**
8. **show snmp mib ifmib ifindex** [*type number*] [**detail**] [**free-list**]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp ifmib ifalias long** **Example:** `Device(config)# snmp ifmib ifalias long`	Configures the Interfaces MIB (IF-MIB) on the system to return ifAlias values of longer than 64 characters to a Network Management System. • If the ifAlias values are not configured using the **snmp ifmib ifalias long** command, the ifAlias description will be restricted to 64 characters.
Step 4	**interface** *type number* **Example:** `Device(config)# interface ethernet 2/4`	Enters interface configuration mode. • The form of this command varies depending on the interface being configured.

	Command or Action	Purpose
Step 5	**description** *text-string* **Example:** `Device(config)# description This text string description can be up to 256 characters long`	Configures a free-text description of the specified interface. • This description can be up to 240 characters in length and is stored as the ifAlias object value in the IF-MIB. • If the ifAlias values are not configured using the **snmp ifmib ifalias long** command, the ifAlias description for SNMP set and get operations is restricted to 64 characters, although the interface description is configured for more than 64 characters by using the **description** command.
Step 6	**end** **Example:** `Device(config)# end`	Exits global configuration mode.
Step 7	**show snmp mib** **Example:** `Device# show snmp mib`	Displays a list of MIB module instance identifiers registered on your system. • The resulting display could be lengthy.
Step 8	**show snmp mib ifmib ifindex** [*type number*] [**detail**] [**free-list**] **Example:** `Device# show snmp mib ifmib ifindex Ethernet 2/0`	Displays the Interfaces MIB ifIndex values registered on your system for all interfaces or the specified interface.

Examples

The following example lists the MIB module instance identifiers registered on your system. The resulting display could be lengthy. Only a small portion is shown here.

```
Device# show snmp mib
system.1
system.2
sysUpTime
system.4
system.5
system.6
system.7
system.8
sysOREntry.2
sysOREntry.3
sysOREntry.4
interfaces.1
ifEntry.1
ifEntry.2
ifEntry.3
ifEntry.4
ifEntry.5
ifEntry.6
ifEntry.7
```

```
ifEntry.8
ifEntry.9
ifEntry.10
ifEntry.11
 --More--
captureBufferEntry.2
captureBufferEntry.3
captureBufferEntry.4
captureBufferEntry.5
captureBufferEntry.6
captureBufferEntry.7
capture.3.1.1
eventEntry.1
eventEntry.2
eventEntry.3
eventEntry.4
eventEntry.5
eventEntry.6
eventEntry.7
logEntry.1
logEntry.2
logEntry.3
logEntry.4
rmon.10.1.1.2
rmon.10.1.1.3
rmon.10.1.1.4
rmon.10.1.1.5
rmon.10.1.1.6
rmon.10.1.1.7
rmon.10.2.1.2
rmon.10.2.1.3
rmon.10.3.1.2
```

The following example shows output for the Interfaces MIB ifIndex values registered on a system for a specific interface:

```
Device# show snmp mib ifmib ifindex Ethernet 2/0
Ethernet2/0: Ifindex = 2
```

The following example shows output for the Interfaces MIB ifIndex values registered on a system for all interfaces:

```
Device# show snmp mib ifmib ifindex
ATM1/0: Ifindex = 1
ATM1/0-aal5 layer: Ifindex = 12
ATM1/0-atm layer: Ifindex = 10
ATM1/0.0-aal5 layer: Ifindex = 13
ATM1/0.0-atm subif: Ifindex = 11
ATM1/0.9-aal5 layer: Ifindex = 32
ATM1/0.9-atm subif: Ifindex = 31
ATM1/0.99-aal5 layer: Ifindex = 36
ATM1/0.99-atm subif: Ifindex = 35
Ethernet2/0: Ifindex = 2
Ethernet2/1: Ifindex = 3
Ethernet2/2: Ifindex = 4
Ethernet2/3: Ifindex = 5
Null0: Ifindex = 14
Serial3/0: Ifindex = 6
Serial3/1: Ifindex = 7
Serial3/2: Ifindex = 8
Serial3/3: Ifindex = 9
```

Troubleshooting Tips

To monitor SNMP trap activity in real time for the purposes of troubleshooting, use the SNMP **debug** commands, including the **debug snmp packet** EXEC command. For documentation of SNMP **debug** commands, see the *Cisco IOS Debug Command Reference.*

Configuration Examples for Circuit Interface Identification Persistence for SNMP

Example Configuring IfAlias Long Name Support

In the following example a long description is applied to the Fast Ethernet interface in slot 1, port adapter 0, and port 0:

```
Device# configure terminal
Device(config)#interface FastEthernet1/0/0
Device(config-if)# description FastEthernet1/0/0 this is a test of a description that exceeds
 64 characters in length
Device(config-if)#ip address 192.168.134.55 255.255.255.0
Device(config-if)#no ip directed-broadcast
Device(config-if)#no ip route-cache distributed
```

Assuming that ifAlias long name support is not yet enabled (the default), the following example shows the results of a mibwalk operation from an NMS:

```
***** SNMP QUERY STARTED *****
 .
 .
 .
 ifXEntry.18.10 (octets) (zero-length)
 ifXEntry.18.11 (octets) Fastethernet1/0/0 this is a test of a description that exceeds 64
 ch
 ifXEntry.18.12 (octets) (zero-length)
 .
 .
 .
```

The following output shows the description that is displayed at the CLI:

```
Device# show interface FastEthernet0/0/0

FastEthernet1/0/0 is administratively down, line protocol is down
  Hardware is Lance, address is 0010.7b4d.7046 (bia 0010.7b4d.7046)
  Description: FastEthernet1/0/0 this is a test of a description that exceeds 64 chh
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
     reliability 252/255, txload 1/255, rxload 1/255
 .
 .
 .
```

In the following example, ifAlias long name support is enabled and the description is displayed again:

```
Device(config)# snmp ifmib ifalias long
Device(config)#interface FastEthernet1/0/0
Device(config-if)# description FastEthernet1/0/0 this is a test of a description that exceeds
64 characters in length
Device(config)#end

Device# show interface FastEthernet1/0/0

FastEthernet1/0/0 is administratively down, line protocol is down
  Hardware is Lance, address is 0010.7b4d.7046 (bia 0010.7b4d.7046)
  Description: FastEthernet1/0/0 this is a test of a description that exceeds 64 characters
 in length
  MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec,
     reliability 252/255, txload 1/255, rxload 1/255
```

```
 .
 .
 .
***** SNMP QUERY STARTED *****
 .
 .
 .
 ifXEntry.18.10 (octets) (zero-length)
 ifXEntry.18.11 (octets) FastEthernet1/0/0 this is a test of a description that exceeds 64
 characters in length
 ifXEntry.18.12 (octets) (zero-length)
 .
 .
 .
```

Example Configuring IfIndex Persistence

The following example shows how to enable IfIndex persistence globally:

```
Device# configure terminal
Device(config)# snmp-server ifindex persist
```

The following example shows how to enable IfIndex persistence on the Ethernet interface:

```
Device# configure terminal
Device(config)# interface ethernet 0/1
Device(config)# snmp-server ifindex persist
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*

Standard/RFC	Title
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*

Standard/RFC	Title
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

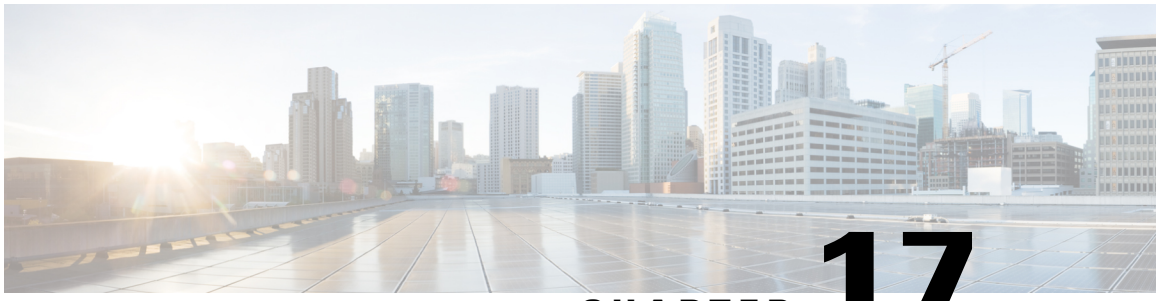
Feature Information for Circuit Interface Identification Persistence for SNMP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 16: Feature Information for Circuit Interface Identification Persistence for SNMP

Feature Name	Releases	Feature Information
Circuit Interface Identification Persistence for SNMP	12.1(3)T 15.0(1)S	The Circuit Interface Identification Persistence for SNMP feature can be used to identify individual circuit-based interfaces for SNMP monitoring.



CHAPTER 17

Interface Index Display for SNMP

The Interface Index Display for SNMP feature introduces new commands and command modifications that allow advanced users of Simple Network Management Protocol (SNMP) to view information about the interface registrations directly on the managed agent. (For the purposes of this document, the agent is the routing device running Cisco software.) In other words, the commands in this feature allow the user to display MIB information from the agent without using an external NMS.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Interface Index Display for SNMP

Interface Index Display for SNMP

The Interface Index Display for SNMP feature introduces new commands and command modifications that allow advanced users of SNMP to view information about interface registrations directly on the managed agent. You can display MIB information from the agent without using an external NMS.

This feature addresses three objects in the Interfaces MIB: ifIndex, ifAlias, and ifName. For complete definitions of these objects, see the IF-MIB.my file available at the Cisco SNMPv2 MIB website.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*

Standard/RFC	Title
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

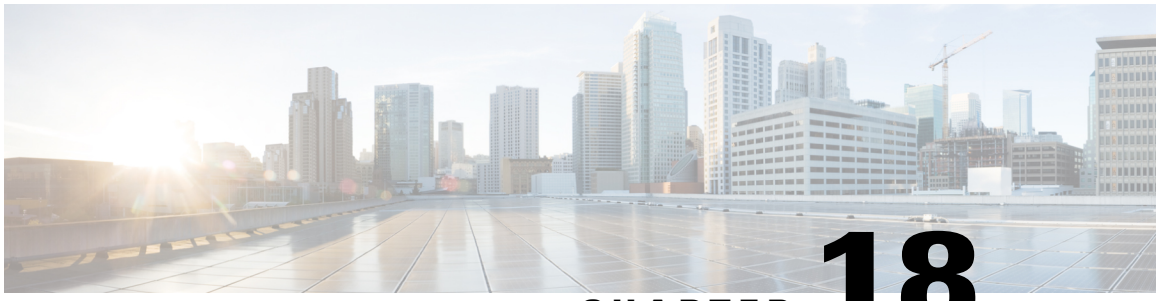
Feature Information for Interface Index Display for SNMP

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 17: Feature Information for Interface Index Display for SNMP

Feature Name	Releases	Feature Information
Interface Index Display for SNMP	12.2(2)T	The Interface Index Display for SNMP feature introduces new commands and command modifications that allow advanced users of SNMP to view information about interface registrations directly on the managed agent. You can display MIB information from the agent without using an external NMS. This feature addresses three objects in the Interfaces MIB: *ifIndex*, *ifAlias*, and *ifName*. For complete definitions of these objects, see the IF-MIB.my file available at the Cisco SNMPv2 MIB website.



CHAPTER 18

Interface Index Persistence

The Interface Index Persistence enhancement allows interfaces to be identified with unique values which will remain constant even when a device is rebooted. These interface identification values are used for network monitoring and management using Simple Network Management Protocol (SNMP).

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information about Interface Index Persistence

Interface Index Persistence

One of the identifiers most commonly used in SNMP-based network management applications is the interface index (ifIndex) value. IfIndex is a unique identifying number associated with a physical or logical interface; as far as most software is concerned, the ifIndex is the “name” of the interface.

Although there is no requirement in the relevant RFCs that the correspondence between particular ifIndex values and their interfaces be maintained across reboots, applications such as device inventory, billing, and fault detection increasingly depend on the maintenance of this correspondence.

This feature adds support for an ifIndex value that can persist across reboots, allowing users to avoid the workarounds previously required for consistent interface identification.

It is currently possible to poll the device at regular intervals to correlate the interfaces to the ifIndex, but it is not practical to poll this interface constantly. If this data is not correlated constantly, however, the data may be made invalid because of a reboot or the insertion of a new card into the device in between polls. Therefore, ifIndex persistence is the only way to guarantee data integrity.

IfIndex persistence means that the mapping between the ifDescr object values and the ifIndex object values (generated from the IF-MIB) will be retained across reboots.

Benefits of Interface Index Persistence

Association of Interfaces with Traffic Targets for Network Management

The Interface Index Persistence feature allows for greater accuracy when collecting and processing network management data by uniquely identifying input and output interfaces for traffic flows and SNMP statistics. Relating each interface to a known entity (such as an ISP customer) allows network management data to be more effectively utilized.

Accuracy for Mediation Fault Detection and Billing

Network data is increasingly being used worldwide for usage-based billing, network planning, policy enforcement, and trend analysis. The ifIndex information is used to identify input and output interfaces for traffic flows and SNMP statistics. Inability to reliably relate each interface to a known entity, such as a customer, invalidates the data.

Configuring Interface Index Persistence

The following sections contain the tasks to configure Interface Index Persistence:

Enabling and Disabling IfIndex Persistence Globally

Perform this task to enable IfIndex persistence globally.

Before You Begin

The configuration tasks described in this section assume that you have configured SNMP on your routing device and are using SNMP to monitor network activity using the Cisco command line interface and/or an NMS application.

The interface-specific ifIndex persistence command (**snmp ifindex persistence**) cannot be used on subinterfaces. A command applied to an interface is automatically applied to all subinterfaces associated with that interface.

Testing indicates that approximately 25 bytes of NVRAM storage are used by this feature per interface. There may be some boot delay exhibited on platforms with lower CPU speeds.

Note After ifIndex persistence commands have been entered, the configuration must be saved using the **copy running-config startup-config** EXEC mode command to ensure consistent ifIndex values.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server ifindex persist**
4. **no snmp-server ifindex persist**
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server ifindex persist** **Example:** `Device(config)# snmp-server ifindex persist`	Globally enables ifIndex values that will remain constant across reboots.
Step 4	**no snmp-server ifindex persist** **Example:** `Device(config)# no snmp-server ifindex persist`	Disables global ifIndex persistence.
Step 5	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Enabling and Disabling IfIndex Persistence on Specific Interfaces

Perform this task to configure ifIndex persistence only on a specific interface.

Tip Use the **snmp ifindex clear** command on a specific interface when you want that interface to use the global configuration setting for ifIndex persistence. This command clears any ifIndex configuration commands previously entered for that specific interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type slot / port*
4. **snmp ifindex persist**
5. **no snmp ifindex persist**
6. **end**
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**interface** *type slot / port* **Example:** `Device(config)# interface FastEthernet 0/1`	Enters interface configuration mode for the specified interface. **Note** Note that the syntax of the interface command will vary depending on the platform you are using.
Step 4	**snmp ifindex persist** **Example:** `Device(config-if)# snmp ifindex persist`	Enables an ifIndex value that is constant across reboots on the specified interface.

	Command or Action	Purpose
Step 5	**no snmp ifindex persist** **Example:** `Device(config-if)# no snmp ifindex persist`	Disables an ifIndex value that is constant across reboots on the specified interface.
Step 6	**end** **Example:** `Device(config-if)# end`	Exits interface configuration mode.
Step 7	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Command Reference
Cisco implementation of RFC 1724, RIP Version 2 MIB Extensions	RIPv2 Monitoring with SNMP Using the RFC 1724 MIB Extensions feature module
DSP Operational State Notifications for notifications to be generated when a digital signaling processor (DSP) is used	DSP Operational State Notifications feature module

Standards and RFCs

Standard/RFC	Title
CBC-DES (DES-56) standard	*Symmetric Encryption Protocol*
STD: 58	*Structure of Management Information Version 2 (SMIv2)*

Standard/RFC	Title
RFC 1067	*A Simple Network Management Protocol*
RFC 1091	*Telnet terminal-type option*
RFC 1098	*Simple Network Management Protocol (SNMP)*
RFC 1157	*Simple Network Management Protocol (SNMP)*
RFC 1213	*Management Information Base for Network Management of TCP/IP-based internets:MIB-II*
RFC 1215	*Convention for defining traps for use with the SNMP*
RFC 1901	*Introduction to Community-based SNMPv2*
RFC 1905	*Common Management Information Services and Protocol over TCP/IP (CMOT)*
RFC 1906	*Telnet X Display Location Option*
RFC 1908	*Simple Network Management Protocol (SNMP)*
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2206	*RSVP Management Information Base using SMIv2*
RFC 2213	*Integrated Services Management Information Base using SMIv2*
RFC 2214	*Integrated Services Management Information Base Guaranteed Service Extensions using SMIv2*
RFC 2271	*An Architecture for Describing SNMP Management Frameworks*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2578	*Structure of Management Information Version 2 (SMIv2)*
RFC 2579	*Textual Conventions for SMIv2*
RFC 2580	*Conformance Statements for SMIv2*
RFC 2981	*Event MIB*
RFC 2982	*Distributed Management Expression MIB*
RFC 3413	*SNMPv3 Applications*

Standard/RFC	Title
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*
RFC 3418	*Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
• Circuit Interface Identification MIB • Cisco SNMPv2 • Ethernet-like Interfaces MIB • Event MIB • Expression MIB Support for Delta, Wildcarding, and Aggregation • Interfaces Group MIB (IF-MIB) • Interfaces Group MIB Enhancements • MIB Enhancements for Universal Gateways and Access Servers • MSDP MIB • NTP MIB • Response Time Monitor MIB • Virtual Switch MIB	To locate and download MIBs for selected platforms, releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Interface Index Persistence

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 18: Feature Information for Interface Index Persistence

Feature Name	Releases	Feature Information
Interface Index Persistence	12.2(15)T 15.0(1)S	The Interface Index Persistence feature allows interfaces to be identified with unique values, which will remain constant even when a device is rebooted. These interface identification values are used for network monitoring and management using SNMP.

Glossary

MPLS VPN —Multiprotocol Label Switching Virtual Private Network

NMS —Network Management System. System responsible for managing at least part of a network. An NMS is generally a reasonably powerful and well-equipped computer, such as an engineering workstation. NMSs communicate with agents to help keep track of network statistics and resources.

SNMP —Simple Network Management Protocol. Network management protocol used almost exclusively in TCP/IP networks. SNMP provides a means to monitor and control network devices and to manage configurations, statistics collection, performance, and security.

SNMP communities —Authentication scheme that enables an intelligent network device to validate SNMP requests.

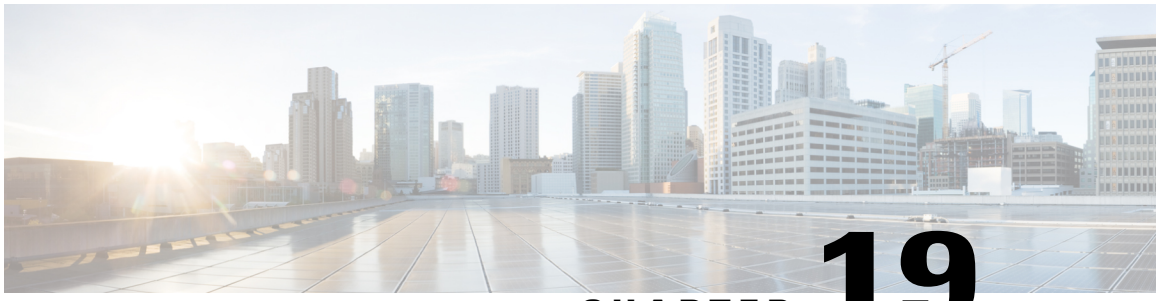
SNMPv2c —Version 2c of the Simple Network Management Protocol. SNMPv2c supports centralized and distributed network management strategies and includes improvements in the Structure of Management Information (SMI), protocol operations, management architecture, and security.

SNMPv3 —Version 3 of the Simple Network Management Protocol. Interoperable standards-based protocol for network management. SNMPv3 provides secure access to devices by a combination of authenticating and encrypting packets over the network.

UDP —User Datagram Protocol. Connectionless transport layer protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams without acknowledgments or guaranteed delivery, requiring that error processing and retransmission be handled by other protocols. UDP is defined in RFC 768.

VRF —A VPN routing and forwarding instance. A VRF consists of an IP routing table, a derived forwarding table, a set of interfaces that use the forwarding table, and a set of rules and routing protocols that determine

what goes into the forwarding table. In general, a VRF includes the routing information that defines a customer VPN site that is attached to a PE device.



CHAPTER 19

SNMP Version 3

The SNMP Version 3 feature provides secure access to devices by authenticating and encrypting data packets over the network. Simple Network Management Protocol version 3 (SNMPv3) is an interoperable, standards-based protocol that is defined in RFCs 3413 to 3415. This module discusses the security features provided in SNMPv3 and describes how to configure the security mechanism to handle SNMP packets.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About SNMP Version 3

Security Features in SNMP Version 3

The security features provided in SNMPv3 are as follows:

- Message integrity—Ensures that a packet has not been tampered with during transit.

- Authentication—Determines that the message is from a valid source.
- Encryption—Scrambles the content of a packet to prevent it from being learned by an unauthorized source.

SNMPv3 is a security model in which an authentication strategy is set up for a user and the group in which the user resides. Security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is used when handling an SNMP packet.

The table below describes the combinations of SNMPv3 security models and levels.

Table 19: SNMP Version 3 Security Levels

Level	Authentication	Encryption	What Happens
noAuthNoPriv	Username	No	Uses a username match for authentication.
authNoPriv	Message Digest Algorithm 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the Hashed Message Authentication Code (HMAC)-MD5 or HMAC-SHA algorithms.
authPriv	MD5 or SHA	Data Encryption Standard (DES)	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. In addition to authentication, provides DES 56-bit encryption based on the Cipher Block Chaining (CBC)-DES (DES-56) standard.

SNMPv3 supports RFCs 1901 to 1908, 2104, 2206, 2213, 2214, and 2271 to 2275. For more information about SNMPv3, see *RFC 2570, Introduction to Version 3 of the Internet-standard Network Management Framework* (this document is not a standard).

Cisco-Specific Error Messages for SNMP Version 3

Simple Network Management Protocol Version 3 (SNMPv3) provides different levels of security. If an authentication or an authorization request fails, a descriptive error message appears to indicate what went wrong. These error messages comply with *RFC 3414, User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*.

You can use the **snmp-server usm cisco** command to disable the descriptive messages, thus preventing malicious users from misusing the information shown in the error messages. The table below describes the Cisco-specific error messages shown when the **snmp-server usm cisco** command is used, and the table compares these messages with the corresponding RFC 3414-compliant error messages.

Table 20: Cisco-Specific Error Messages for SNMPv3

Configured Security Level	Security Level of Incoming SNMP Message	RFC 3414-Compliant Error Indication	Cisco-Specific Error Messages
noAuthNoPriv	noAuthNoPriv	No error	No error
	authNoPriv	unsupportedSecurityLevel	unknownUserName
	authPriv	unsupportedSecurityLevel	unknownUserName
authNoPriv	noAuthNoPriv	AUTHORIZATION_ERROR	unknownUserName
	authNoPriv with correct authentication password	No error	No error
	authNoPriv with incorrect authentication password	wrongDigests	unknownUserName
	authPriv	unsupportedSecurityLevel	unknownUserName
authPriv	noAuthNoPriv	AUTHORIZATION_ERROR	unknownUserName
	authNoPriv with correct authentication password	AUTHORIZATION_ERROR	unknownUserName
	authNoPriv with incorrect authentication password	AUTHORIZATION_ERROR	unknownUserName
	authPriv with correct authentication password and correct privacy password	No error	No error
	authPriv with correct authentication password and incorrect privacy password	No response	No response
	authPriv with incorrect authentication password and correct privacy password	wrongDigests	unknownUserName
	authPriv with incorrect authentication password and incorrect privacy password	wrongDigests	unknownUserName

Note If an SNMP user belonging to an SNMP group is not configured with the password or if the group security level is not the same as the user security level, the error shown is “AUTHORIZATION_ERROR”. The Cisco-specific error message for this scenario is “unknownUserName”.

How to Configure SNMP Version 3

To configure the Simple Network Management Protocol Version 3 (SNMPv3) security mechanism and to use it to handle SNMP packets, you must configure SNMP groups and users with passwords.

Configuring the SNMP Server

To configure an SNMP server user, specify an SNMP group or a table that maps SNMP users to SNMP views. Then, specify the IP address or port number for the remote SNMP agent of the device where the user resides. Also, before you configure remote users for a particular agent, configure the SNMP engine ID by using the **snmp-server engineID** command for the remote agent. The SNMP engine ID of the remote agent is required to compute the authentication or privacy digests for the SNMP password. If the remote engine ID is not configured first, the configuration command will fail.

SNMP passwords are localized using the SNMP engine ID of the authoritative SNMP engine. For SNMP notifications such as inform requests, the authoritative SNMP agent is the remote agent. You must configure the SNMP engine ID of the remote agent in the SNMP database before you can send proxy requests or inform requests to it.

Note The SNMP user cannot be removed if the engine ID is changed after configuring the SNMP user. To remove the user, you must first reconfigure all the SNMP configurations.

Note Default values do not exist for authentication or privacy algorithms when you configure the SNMP commands. Also, no default passwords exist. The minimum length for a password is one character, although it is recommended to use at least eight characters for security. If you forget a password, you cannot recover it and must reconfigure the user. You can specify either a plain text password or a localized MD5 digest.

Perform this task to specify an SNMP server group name and to add a new user to an SNMP group.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server group** [*group-name* {**v1** | **v2c** | **v3** [**auth** | **noauth** | **priv**]}] [**read** *read-view*] [**write** *write-view*] [**notify** *notify-view*] [**access** *access-list*]
4. **snmp-server engineID** {**local** *engine-id* | **remote** *ip-address* [**udp-port** *udp-port-number*] [**vrf** *vrf-name*] *engine-id-string*}
5. **snmp-server user** *user-name group-name* [**remote** *ip-address* [**udp-port** *port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**access** *access-list*]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server group** [*group-name* {**v1** | **v2c** | **v3** [**auth** | **noauth** | **priv**]}] [**read** *read-view*] [**write** *write-view*] [**notify** *notify-view*] [**access** *access-list*] **Example:** `Device(config)# snmp-server group group1 v3 auth access lmnop`	Configures the SNMP server group to enable authentication for members of a specified named access list. • In this example, the SNMP server group group1 is configured to enable user authentication for members of the named access list lmnop.
Step 4	**snmp-server engineID** {**local** *engine-id* | **remote** *ip-address* [**udp-port** *udp-port-number*] [**vrf** *vrf-name*] *engine-id-string*} **Example:** `Device(config)# snmp-server engineID remote 172.16.15.4 udp-port 120 1a2833c0129a`	Configures the SNMP engine ID. • In this example, the SNMP engine ID is configured for a remote user.
Step 5	**snmp-server user** *user-name group-name* [**remote** *ip-address* [**udp-port** *port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**access** *access-list*]	Adds a new user to an SNMPv3 group and configures a plain text password for the user. **Note** For the *auth-password* argument, the minimum length is one character; the recommended length is at least eight characters, and the password should include both letters and numbers.

	Command or Action	Purpose
	Example: `Device(config)# snmp-server user user1 group1 v3 auth md5 password123`	**Note** If you have the localized MD5 or SHA digest, you can specify the digest instead of the plain text password. The digest should be formatted as aa:bb:cc:dd, where aa, bb, cc, and dd are hexadecimal values. Also, the digest should be exactly 16 octets in length.
Step 6	**end** **Example:** `Device(config)# end`	Exits global configuration mode.

Verifying SNMP Version 3

Perform this task to verify the Simple Network Management Protocol Version 3 (SNMPv3) configuration. The **show** commands can be entered in any order.

SUMMARY STEPS

1. **enable**
2. **show snmp group**
3. **show snmp user** [*username*]
4. **show snmp engineID**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enables privileged EXEC mode. • Enter your password if prompted.

Step 2	**show snmp group**
	Example:
	`Device# show snmp group`
	`groupname: V1 security model:v1`
	`readview : v1default writeview: <no writeview specified>`
	`notifyview: <no notifyview specified>`
	`row status: active`
	`groupname: ILMI security model:v1`
	`readview : *ilmi writeview: *ilmi`
	`notifyview: <no notifyview specified>`
	`row status: active`
		Displays information about each SNMP group in the network.
		Displays information about each SNMP group in the network.

	Command or Action	Purpose

	`groupname: ILMI security model:v2c`
	`readview : *ilmi writeview: *ilmi`
	
	`notifyview: <no notifyview specified>`
	`row status: active`
	`groupname: group1 security model:v1`
	`readview : v1default writeview: <no writeview specified>`
	`notifyview: <no notifyview specified>`
	`row status: active`	

Step 3	**show snmp user** [*username*]
	
	Example:
	
	`Device# show snmp user user1`
	
	`User name: user1`
	`Engine ID: 000000090200000000C025808`
	`storage-type: nonvolatile active access-list: 10`
	`Rowstatus: active`
	`Authentication Protocol: MD5`
	`Privacy protocol: DES`
	`Group name: group1`
		Displays information about configured characteristics of an SNMP user.

Step 4	**show snmp engineID**
	
	Example:
	`Device# show snmp engineID`
	
	`Local SNMP engineID: 1A2836C0129A`
	`Remote Engine ID IP-addr Port`
	`1A2833C0129A remote 10.2.28.1 120`
		Displays information about the SNMP engine ID that is configured for an SNMP user.

Configuration Examples for SNMP Version 3

Example: Configuring SNMP Version 3

The following example shows how to enable Simple Network Management Protocol Version 3 (SNMPv3). The configuration permits any SNMP manager to access all objects with read-only permissions using the community string named “public”. This configuration does not cause the device to send traps.

```
Device(config)# snmp-server community public
```

The following example shows how to configure a remote user to receive traps at the “noAuthNoPriv” security level when the SNMPv3 security model is enabled:

```
Device(config)# snmp-server group group1 v3 noauth
Device(config)# snmp-server user remoteuser1 group1 remote 10.12.8.4
Device(config)# snmp-server host 10.12.8.4 informs version 3 noauth remoteuser config
```

The following example shows how to configure a remote user to receive traps at the “authNoPriv” security level when the SNMPv3 security model is enabled:

```
Device(config)# snmp-server group group2 v3 auth
Device(config)# snmp-server user AuthUser group2 remote 10.12.8.4 v3 auth md5 password1
```

The following example shows how to configure a remote user to receive traps at the “priv” security level when the SNMPv3 security model is enabled:

```
Device(config)# snmp-server group group3 v3 priv
Device(config)# snmp-server user PrivateUser group3 remote 10.12.8.4 v3 auth md5 password1
 priv access des56
```

Additional References for SNMP Version 3

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
SNMP commands: complete command syntax, command mode, command history, defaults, usage guidelines, and examples	Cisco IOS SNMP Support Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 2104	*HMAC: Keyed-Hashing for Message Authentication*
RFC 2570	*Introduction to Version 3 of the Internet-standard Network Management Framework*
RFC 2576	*Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*
RFC 3413	*SNMPv3 Applications*
RFC 3414	*User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*
RFC 3415	*View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*

MIBs

MIB	MIBs Link
SNMP-COMMUNITY-MIB	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

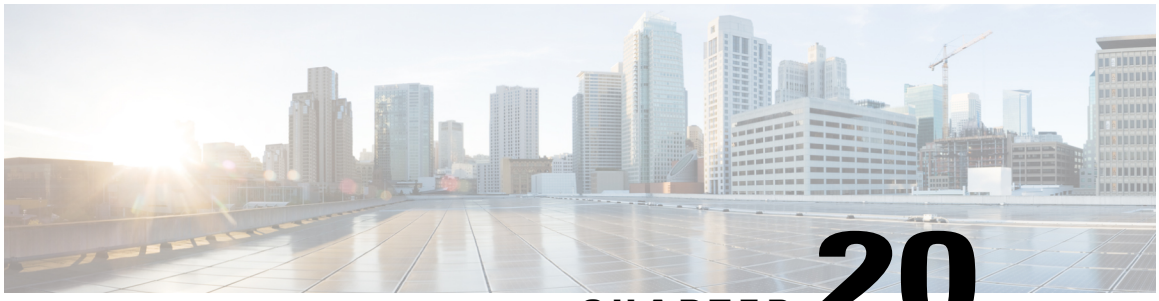
Feature Information for SNMP Version 3

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 21: Feature Information for SNMP Version 3

Feature Name	Releases	Feature Information
SNMP Version 3		The SNMP Version 3 feature is used to provide secure access to devices by authenticating and encrypting data packets over the network.



CHAPTER 20

AES and 3-DES Encryption Support for SNMP Version 3

The AES and 3-DES Encryption Support for SNMP Version 3feature enhances the encryption capabilities of SNMP version 3. This support for Simple Network Management Protocol (SNMP) version 3 User-Based Security Model (USM) is complaint with RFC 3414, which defines DES as the only required method of message encryption for SNMP version 3 authPriv mode.

The AES and 3-DES Encryption Support for SNMP Version 3 feature adds Advanced Encryption Standard (AES) 128-bit encryption in compliance with RFC 3826. RFC 3826 extensions have been included in the SNMP-USM-AES-MIB. In addition, Cisco-specific extensions to support Triple-Data Encryption Algorithm (3-DES) and AES 192-bit and 256-bit encryption have been added to the CISCO-SNMP-USM-MIB. Additional information can be found in the Internet-Draft titled *Extension to the User-Based Security Model (USM) to Support Triple-DES EDE in "Outside" CBC Mode,* which can be found at the following URL: http://www.snmp.com/eso/draft-reeder-snmpv3-usm-3desede-00.txt.

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see Bug Search Tool and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for AES and 3-DES Encryption Support for SNMP Version 3

- The network management station (NMS) must support SNMP version 3 to use this feature of the SNMP agent.
- This feature is available in Cisco IOS XE software images where encryption algorithms are supported.

Information About AES and 3-DES Encryption Support for SNMP Version 3

SNMP Architecture

The architecture for describing Internet Management Frameworks contained in RFC 3411 describes the SNMP engine as composed of the following components:

- Dispatcher
- Message Processing Subsystem
- Security Subsystem
- Access Control Subsystem

Applications make use of the services of these subsystems. It is important to understand the SNMP architecture and the terminology of the architecture to understand where the Security Model fits into the architecture and interacts with the other subsystems within the architecture. The information is contained in RFC 3411 and your are encouraged to review this RFC to obtain an understanding of the SNMP architecture and subsystem interactions.

Encryption Key Support

In the AES and 3-DES Encryption Support for SNMP Version 3 feature the Cipher Block Chaining/Data Encryption Standard (CBC-DES) is the privacy protocol. Originally only DES was supported (as per RFC 3414). This feature adds support for AES-128 (as per RFC 3826) and AES-192, AES-256 and 3-DES (as per CISCO-SNMP-USM-OIDS-MIB).

- AES encryption uses the Cipher Feedback (CFB) mode with encryption key sizes of 128, 192, or 256 bits.
- 3DES encryption uses the 168-bit key size for encryption.

The AES Cipher Algorithm in the SNMP User-based Security Model draft describes the use of AES with 128-bit key size. However, the other options are also implemented with the extension to use the USM. There is currently no standard for generating localized keys for 192- or 256-bit size keys for AES or for 168-bit size key for 3-DES. There is no authentication protocol available with longer keys.

Management Information Base Support

The AES and 3-DES Encryption Support for SNMP Version 3AES and 3-DES Encryption Support for SNMP Version 3 feature supports the selection of privacy protocols through the CLI and the Management Information Base (MIB). A new standard MIB, SNMP-USM-AES-MIB, provides support for the 128-bit key in AES. The extended options of AES with 192- or 256-bit keys and 3-DES are supported as extensions to the SNMP-USM-MIB, in the Cisco-specific MIB, CISCO-SNMP-USM-OIDS-MIB.

How to Configure AES and 3-DES Encryption Support for SNMP Version 3

Adding a New User to an SNMP Group

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **snmp-server user** *username group-name* [**remote** *host* [**udp-port** *port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**priv** {**des** | **3des** | **aes {128 | 192 |256}**} *privpassword*] [**access** [**ipv6** *nacl*] {*acl-number* | *acl-name*}]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Device> enable`	Enters privileged EXEC mode. • Enter your password when prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**snmp-server user** *username group-name* [**remote** *host* [**udp-port** *port*]] {**v1** | **v2c** | **v3** [**encrypted**] [**auth** {**md5** | **sha**} *auth-password*]} [**priv** {**des** | **3des** | **aes {128 | 192 |256}**} *privpassword*] [**access** [**ipv6** *nacl*] {*acl-number* | *acl-name*}]	Adds an SNMP user, specifies a group to which the user belongs, specifies the authorization algorithm to be used (MD5 or SHA), specifies the privacy algorithm to be used (DES, 3-DES, AES, AES-192, or AES-256), and specifies the password to be associated with this privacy protocol.

	Command or Action	Purpose
	Example: `Device(config)# snmp-server user new-user new-group v3 auth md5 secureone priv aes 128 privatetwo access 2`	

Verifying SNMP User Configuration

To display information about the configured characteristics of Simple Network Management Protocol (SNMP) users, use the **show snmp user** command in privileged EXEC mode.

Note The **show snmp user** command displays all the users configured on the router. However, unlike other SNMP configurations, the **snmp-server user** command will not appear on the "show running" output.

SUMMARY STEPS

1. **enable**
2. **show snmp user** [*username*

DETAILED STEPS

Step 1 **enable**
Enters privileged EXEC mode. Enter your password when prompted.

Step 2 **show snmp user** [*username*
The following example specifies the username as abcd, the engine ID string as 00000090200000000C025808, and the storage type as nonvolatile:

Example:

```
Device# show snmp user
 abcd
User name: abcd
Engine ID: 00000090200000000C025808
storage-type: nonvolatile       active access-list: 10
Rowstatus: active
Authentication Protocol: MD5
Privacy protocol: 3DES
Group name: VacmGroupName
Group name: VacmGroupName
```

Additional References

Related Documents

Related Topic	Document Title
Cisco software commands	Cisco IOS Master Command List, All Releases
Cisco Network Management commands	*Cisco IOS Network Management Command Reference*
SNMP configuration	" Configuring SNMP Support " chapter in the *Cisco Network Management Configuration Guide*
SNMP Support for VPNs	SNMP Notification Support for VPNs

Standards

Standard	Title
None	--

MIBs

MIB	MIBs Link
• CISCO-PING-MIB • IP-FORWARD-MIB • SNMP-VACM-MIB, *The View-based Access Control Model (ACM) MIB for SNMP*	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFC	Title
RFC 1441	*Introduction to version 2 of the Internet-standard Network Management Framework*
RFC 1442	*Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1443	*Textual Conventions for version 2 of the Simple Network Management Protocol (SNMPv2)*

RFC	Title
RFC 1444	*Conformance Statements for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1445	*Administrative Model for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1446	*Security Protocols for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1447	*Party MIB for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1448	*Protocol Operations for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1449	*Transport Mappings for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 1450	*Management Information Base for version 2 of the Simple Network Management Protocol (SNMPv2)*
RFC 2571	*An Architecture for Describing SNMP Management Frameworks*
RFC 2576	*Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for AES and 3-DES Encryption Support for SNMP Version 3

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to . An account on Cisco.com is not required.

Table 22: Feature Information for AES and 3-DES Encryption Support for SNMP Version 3

Feature Name	Releases	Feature Information
AES and 3-DES Encryption Support for SNMP Version 3		The AES and 3-DES Encryption Support for SNMP Version 3 feature enhances the encryption capabilities of SNMP version 3. This support for SNMP version 3 USM is complaint with RFC 3414, which defines DES as the only required method of message encryption for SNMP version 3 authPriv mode. The AES and 3-DES Encryption Support for SNMP Version 3 feature adds AES 128-bit encryption in compliance with RFC 3826.