CHAPTER 36

Configuring SNMP

Finding Feature Information

Your software release may not support all the features documented in this chapter. For the latest feature information and caveats, see the release notes for your platform and software release.

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

Prerequisites for SNMP

An SNMP *group* is a table that maps SNMP users to SNMP views. An SNMP *user* is a member of an SNMP group. An SNMP *host* is the recipient of an SNMP trap operation. An SNMP *engine ID* is a name for the local or remote SNMP engine.

- If the switch starts and the switch startup configuration has at least one **snmp-server** global configuration command, the SNMP agent is enabled.
- When configuring an SNMP group, do not specify a notify view. The **snmp-server host** global configuration command autogenerates a notify view for the user and then adds it to the group associated with that user. Modifying the group's notify view affects all users associated with that group. See the *Cisco IOS Network Management Command Reference* for information about when you should configure notify views.
- To configure a remote user, specify the IP address or port number for the remote SNMP agent of the device where the user resides.
- Before you configure remote users for a particular agent, configure the SNMP engine ID, using the **snmp-server engineID** global configuration with the **remote** option. The remote agent's SNMP engine ID and user password are used to compute the authentication and privacy digests. If you do not configure the remote engine ID first, the configuration command fails.

Restrictions for SNMP

- When configuring SNMP informs, you need to configure the SNMP engine ID for the remote agent in the SNMP database before you can send proxy requests or informs to it.
- If a local user is not associated with a remote host, the switch does not send informs for the **auth** (authNoPriv) and the **priv** (authPriv) authentication levels.

- Changing the value of the SNMP engine ID has important implications. A user's password (entered on the command line) is converted to an MD5 or SHA security digest based on the password and the local engine ID. The command-line password is then destroyed, as required by RFC 2274. Because of this deletion, if the value of the engine ID changes, the security digests of SNMPv3 users become invalid, and you need to reconfigure SNMP users by using the **snmp-server user** *username* global configuration command. Similar restrictions require the reconfiguration of community strings when the engine ID changes.

Information About SNMP

SNMP

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between managers and agents. The SNMP system consists of an SNMP manager, an SNMP agent, and a MIB. The SNMP manager can be part of a network management system (NMS) such as CiscoWorks. The agent and MIB reside on the switch. To configure SNMP on the switch, you define the relationship between the manager and the agent.

The SNMP agent contains MIB variables whose values the SNMP manager can request or change. A manager can get a value from an agent or store a value into the agent. The agent gathers data from the MIB, the repository for information about device parameters and network data. The agent can also respond to a manager's requests to get or set data.

An agent can send unsolicited traps to the manager. Traps are messages alerting the SNMP manager to a condition on the network. Traps can mean improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

SNMP Versions

This software release supports these SNMP versions:

- SNMPv1—The Simple Network Management Protocol, a Full Internet Standard, defined in RFC 1157.
- SNMPv2C replaces the Party-based Administrative and Security Framework of SNMPv2Classic with the community-string-based Administrative Framework of SNMPv2C while retaining the bulk retrieval and improved error handling of SNMPv2Classic. It has these features:
  - SNMPv2—Version 2 of the Simple Network Management Protocol, a Draft Internet Standard, defined in RFCs 1902 through 1907.
  - SNMPv2C—The community-string-based Administrative Framework for SNMPv2, an Experimental Internet Protocol defined in RFC 1901.
- SNMPv3—Version 3 of the SNMP is an interoperable standards-based protocol defined in RFCs 2273 to 2275. SNMPv3 provides secure access to devices by authenticating and encrypting packets over the network and includes these security features:
  - Message integrity—Ensures that a packet was not tampered with in transit.
  - Authentication—Determines that the message is from a valid source.

– Encryption—Mixes the contents of a package to prevent it from being read by an unauthorized source.

Note To select encryption, enter the **priv** keyword. This keyword is available only when the cryptographic (encrypted) software image is installed.

Both SNMPv1 and SNMPv2C use a community-based form of security. The community of managers able to access the agent's MIB is defined by an IP address access control list and password.

SNMPv2C includes a bulk retrieval mechanism and more detailed error message reporting to management stations. The bulk retrieval mechanism retrieves tables and large quantities of information, minimizing the number of round-trips required. The SNMPv2C improved error-handling includes expanded error codes that distinguish different kinds of error conditions; these conditions are reported through a single error code in SNMPv1. Error return codes in SNMPv2C report the error type.

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy set up for a user and the group within which the user resides. A security level is the permitted level of security within a security model. A combination of the security level and the security model determine which security mechanism is used when handling an SNMP packet. Available security models are SNMPv1, SNMPv2C, and SNMPv3.

Table 36-1 identifies the characteristics of the different combinations of security models and levels.

Table 36-1 SNMP Security Models and Levels

Model	Level	Authentication	Encryption	Result
SNMPv1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv2C	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
SNMPv3	noAuthNoPriv (requires the LAN Base image)	Username	No	Uses a username match for authentication.
SNMPv3	authNoPriv (requires the LAN Base image)	Message Digest 5 (MD5) or Secure Hash Algorithm (SHA)	No	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
SNMPv3	authPriv (requires the LAN Base image)	MD5 or SHA	Data Encryption Standard (DES) or Advanced Encryption Standard (AES)	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Allows specifying the User-based Security Model (USM) with these encryption algorithms: • DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard. • 3DES 168-bit encryption • AES 128-bit, 192-bit, or 256-bit encryption

You must configure the SNMP agent to use the SNMP version supported by the management station. Because an agent can communicate with multiple managers, you can configure the software to support communications using SNMPv1, SNMPv2C, or SNMPv3.

SNMP Manager Functions

The SNMP manager uses information in the MIB to perform the operations described in Table 36-2.

Table 36-2 SNMP Operations

Operation	Description
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table.[1]
get-bulk-request[2]	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by an NMS.
set-request	Stores a value in a specific variable.
trap	An unsolicited message sent by an SNMP agent to an SNMP manager when some event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.
2. The **get-bulk** command only works with SNMPv2 or later.

SNMP Agent Functions

The SNMP agent responds to SNMP manager requests as follows:

- Get a MIB variable—The SNMP agent begins this function in response to a request from the NMS. The agent retrieves the value of the requested MIB variable and responds to the NMS with that value.
- Set a MIB variable—The SNMP agent begins this function in response to a message from the NMS. The SNMP agent changes the value of the MIB variable to the value requested by the NMS.

The SNMP agent also sends unsolicited trap messages to notify an NMS that a significant event has occurred on the agent. Examples of trap conditions include, but are not limited to, when a port or module goes up or down, when spanning-tree topology changes occur, and when authentication failures occur.

SNMP Community Strings

SNMP community strings authenticate access to MIB objects and function as embedded passwords. In order for the NMS to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch.

A community string can have one of these attributes:

- Read-only (RO)—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access.
- Read-write (RW)—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings.

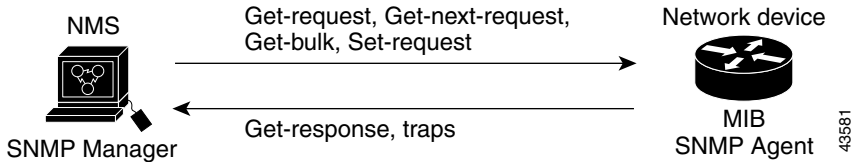
When a cluster is created, the command switch manages the exchange of messages among member switches and the SNMP application. If you are using CNA, it appends the member switch number (*@esN*, where *N* is the switch number) to the first configured RW and RO community strings on the command switch and propagates them to the member switches. For more information, see Chapter 6, "Configuring Switch Clusters" and see *Getting Started with Cisco Network Assistant*, available on Cisco.com.

Using SNMP to Access MIB Variables

An example of an NMS is the CiscoWorks network management software. CiscoWorks 2000 software uses the switch MIB variables to set device variables and to poll devices on the network for specific information. The results of a poll can be displayed as a graph and analyzed to troubleshoot internetworking problems, increase network performance, verify the configuration of devices, monitor traffic loads, and more.

As shown in Figure 36-1, the SNMP agent gathers data from the MIB. The agent can send traps, or notification of certain events, to the SNMP manager, which receives and processes the traps. Traps alert the SNMP manager to a condition on the network such as improper user authentication, restarts, link status (up or down), MAC address tracking, and so forth. The SNMP agent also responds to MIB-related queries sent by the SNMP manager in *get-request*, *get-next-request*, and *set-request* format.

Figure 36-1 SNMP Network



SNMP Notifications

SNMP allows the switch to send notifications to SNMP managers when particular events occur. SNMP notifications can be sent as traps or inform requests. In command syntax, unless there is an option in the command to select either traps or informs, the keyword **traps** refers to either traps or informs, or both. Use the **snmp-server host** command to specify whether to send SNMP notifications as traps or informs.

Note SNMPv1 does not support informs.

Traps are unreliable because the receiver does not send an acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. When an SNMP manager receives an inform request, it acknowledges the message with an SNMP response protocol data unit (PDU). If the sender does not receive a response, the inform request can be sent again. Because they can be resent, informs are more likely than traps to reach their intended destination.

The characteristics that make informs more reliable than traps also consume more resources in the switch and in the network. Unlike a trap, which is discarded as soon as it is sent, an inform request is held in memory until a response is received or the request times out. Traps are sent only once, but an inform might be resent or retried several times. The retries increase traffic and contribute to a higher overhead on the network. Therefore, traps and informs require a trade-off between reliability and resources. If it is important that the SNMP manager receive every notification, use inform requests. If traffic on the network or memory in the switch is a concern and notification is not required, use traps.

SNMP ifIndex MIB Object Values

In an NMS, the IF-MIB generates and assigns an interface index (ifIndex) object value that is a unique number greater than zero to identify a physical or a logical interface. When the switch reboots or the switch software is upgraded, the switch uses this same value for the interface. For example, if the switch assigns a port 2 an ifIndex value of 10003, this value is the same after the switch reboots.

The switch uses one of the values in Table 36-3 to assign an ifIndex value to an interface.

Table 36-3 ifIndex Values

Interface Type	ifIndex Range
SVI	1–4999
EtherChannel	5001–5048
Physical (such as Gigabit Ethernet or SFP-module interfaces) based on type and port numbers	10000–14500
Null	10501
Loopback and Tunnel	24567 +

Note The switch might not use sequential values within a range.

Community Strings

You use the SNMP community string to define the relationship between the SNMP manager and the agent. The community string acts like a password to permit access to the agent on the switch. Optionally, you can specify one or more of these 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

SNMP Notifications

A trap manager is a management station that receives and processes traps. Traps are system alerts that the switch generates when certain events occur. By default, no trap manager is defined, and no traps are sent. Switches running this Cisco IOS release can have an unlimited number of trap managers.

Note Many commands use the word *traps* in the command syntax. Unless there is an option in the command to select either traps or informs, the keyword **traps** refers to traps, informs, or both. Use the **snmp-server host** global configuration command to specify whether to send SNMP notifications as traps or informs.

This table describes the supported switch traps (notification types). You can enable any or all of these traps and configure a trap manager to receive them. To enable the sending of SNMP inform notifications, use the **snmp-server enable traps** global configuration command combined with the **snmp-server host** *host-addr* **informs** global configuration command.

Table 36-4 Switch Notification Types

Notification Type Keyword	Description
bridge	Generates STP bridge MIB traps.
config	Generates a trap for SNMP configuration changes.
copy-config	Generates a trap for SNMP copy configuration changes.
entity	Generates a trap for SNMP entity changes.
cpu threshold	Allows CPU-related traps.
envmon	Generates environmental monitor traps. You can enable any or all of these environmental traps: fan, shutdown, status, supply, temperature.
errdisable	Generates a trap for an error-disabled VLAN port. You can also set a maximum trap rate per minute. The range is from 0 to 10000; the default is 0, which means there is no rate limit.
flash	Generates SNMP FLASH notifications.
hsrp	Generates a trap for Hot Standby Router Protocol (HSRP) changes.
ipmulticast	Generates a trap for IP multicast routing changes.
mac-notification	Generates a trap for MAC address notifications.
msdp	Generates a trap for Multicast Source Discovery Protocol (MSDP) changes.
ospf	Generates a trap for Open Shortest Path First (OSPF) changes. You can enable any or all of these traps: Cisco specific, errors, link-state advertisement, rate limit, retransmit, and state changes.
pim	Generates a trap for Protocol-Independent Multicast (PIM) changes. You can enable any or all of these traps: invalid PIM messages, neighbor changes, and rendezvous point (RP)-mapping changes.
port-security	Generates SNMP port security traps. You can also set a maximum trap rate per second. The range is from 0 to 1000; the default is 0, which means that there is no rate limit. **Note** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: • **snmp-server enable traps port-security** • **snmp-server enable traps port-security trap-rate** *rate*
rtr	Generates a trap for the SNMP Response Time Reporter (RTR).
snmp	Generates a trap for SNMP-type notifications for authentication, cold start, warm start, link up or link down.
storm-control	Generates a trap for SNMP storm control. You can also set a maximum trap rate per minute. The range is from 0 to 1000; the default is 0 (no limit is imposed; a trap is sent at every occurrence).
stpx	Generates SNMP STP Extended MIB traps.
syslog	Generates SNMP syslog traps.
tty	Generates a trap for TCP connections. This trap is enabled by default.
vlan-membership	Generates a trap for SNMP VLAN membership changes.
vlancreate	Generates SNMP VLAN created traps.

Table 36-4 Switch Notification Types (continued)

Notification Type Keyword	Description
vlandelete	Generates SNMP VLAN deleted traps.
vtp	Generates a trap for VLAN Trunking Protocol (VTP) changes.

Note Though visible in the command-line help strings, the **fru-ctrl, insertion**, and **removal** keywords are not supported.

You can use the **snmp-server host** global configuration command to a specific host to receive the notification types listed in Table 36-4.

Default SNMP Settings

Table 36-5 Default SNMP Settings

Feature	Default Setting
SNMP agent	Disabled[1].
SNMP trap receiver	None configured.
SNMP traps	None enabled except the trap for TCP connections (**tty**).
SNMP version	If no **version** keyword is present, the default is Version 1.
SNMPv3 authentication	If no keyword is entered, the default is the **noauth** (noAuthNoPriv) security level.
SNMP notification type	If no type is specified, all notifications are sent.

1. This is the default when the switch starts and the startup configuration does not have any **snmp-server** global configuration commands.

How to Configure SNMP

Disabling the SNMP Agent

The **no snmp-server** global configuration command disables all running versions (Version 1, Version 2C, and Version 3) on the device. No specific Cisco IOS command exists to enable SNMP. The first **snmp-server** global configuration command that you enter enables all versions of SNMP.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**no snmp-server**	Disables the SNMP agent operation.
Step 3	**end**	Returns to privileged EXEC mode.

Configuring Community Strings

Note To disable access for an SNMP community, set the community string for that community to the null string (do not enter a value for the community string).

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**snmp-server community** *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]	Configures the community string. **Note** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. • *string*—Specifies a string that acts like a password and permits access to the SNMP protocol. You can configure one or more community strings of any length. • (Optional) **view**—Specifies the view record accessible to the community. • (Optional) Specifies either read-only (**ro**) if you want authorized management stations to retrieve MIB objects, or specifies read-write (**rw**) if you want authorized management stations to retrieve and modify MIB objects. By default, the community string permits read-only access to all objects. • (Optional) *access-list-number*—Specifies an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	(Optional) If you specified an IP standard access list number in Step 2, then create the list, repeating the command as many times as necessary. • *access-list-number*—Specifies the access list number specified in Step 2. • **deny** — Denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • *source*—Specifies the IP address of the SNMP managers that are permitted to use the community string to gain access to the agent. • (Optional) *source-wildcard*—Specifies the wildcard bits in dotted decimal notation to be applied to the source. Place ones in the bit positions that you want to ignore. The access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Returns to privileged EXEC mode.

Configuring SNMP Groups and Users

You can specify an identification name (engine ID) for the local or remote SNMP server engine on the switch. You can configure an SNMP server group that maps SNMP users to SNMP views, and you can add new users to the SNMP group.

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**snmp-server engineID** {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}	Configures a name for either the local or remote copy of SNMP. • The *engineid-string* is a 24-character ID string with the name of the copy of SNMP. You need not specify the entire 24-character engine ID if it has trailing zeros. Specify only the portion of the engine ID up to the point where only zeros remain in the value. For example, to configure an engine ID of 123400000000000000000000, you can enter this: **snmp-server engineID local 1234** • If you select **remote**, specify the *ip-address* of the device that contains the remote copy of SNMP and the optional User Datagram Protocol (UDP) port on the remote device. The default is 162.

	Command	Purpose
Step 3	**snmp-server group** *groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configures a new SNMP group on the remote device. • *groupname*—Specifies the name of the group. • Specify a security model: – **v1** is the least secure of the possible security models. – **v2c** is the second least secure model. It allows transmission of informs and integers twice the normal width. – **v3,** the most secure, requires you to select an authentication level: **auth**—Enables the Message Digest 5 (MD5) and the Secure Hash Algorithm (SHA) packet authentication. **noauth**—Enables the noAuthNoPriv security level. This is the default if no keyword is specified. **priv**—Enables Data Encryption Standard (DES) packet encryption (also called *privacy*). **Note** The **priv** keyword is available only when the cryptographic software image is installed. • (Optional) **read** *readview*—Specifies a string (not to exceed 64 characters) that is the name of the view in which you can only view the contents of the agent. • (Optional) **write** *writeview*—Specifies a string (not to exceed 64 characters) that is the name of the view in which you enter data and configure the contents of the agent. • (Optional) **notify** *notifyview*—Specifies a string (not to exceed 64 characters) that is the name of the view in which you specify a notify, inform, or trap. • (Optional) **access** *access-list*—Specifies a string (not to exceed 64 characters) that is the name of the access list.

	Command	Purpose
Step 4	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]} [*priv* {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]	Adds a new user for an SNMP group. • *username*—Specifies a name of the user on the host that connects to the agent. • *groupname*—Specifies a name of the group to which the user is associated. • **remote**—Specifies a remote SNMP entity to which the user belongs and the hostname or IP address of that entity with the optional UDP port number. The default is 162. • Enters the SNMP version number (**v1**, **v2c**, or **v3**). If you enter **v3**, you have these additional options: – **encrypted**—Specifies that the password appears in encrypted format. This keyword is available only when the **v3** keyword is specified. – **auth**—Specifies an authentication level setting session that can be either the HMAC-MD5-96 (**md5**) or the HMAC-SHA-96 (**sha**) authentication level and requires a password string *auth-password* (not to exceed 64 characters). • If you enter **v3** and the switch is running the cryptographic software image, you can also configure a private (**priv**) encryption algorithm and password string *priv-password* (not to exceed 64 characters). – **priv**—Specifies the User-based Security Model (USM). – **des**—Specifies the use of the 56-bit DES algorithm. – **3des**—Specifies the use of the 168-bit DES algorithm. – **aes**—Specifies the use of the DES algorithm. You must select either 128-bit, 192-bit, or 256-bit encryption. • (Optional) Enters **access** *access-list* with a string (not to exceed 64 characters) that is the name of the access list.
Step 5	**end**	Returns to privileged EXEC mode.

Configuring SNMP Notifications

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**snmp-server engineID remote** *ip-address engineid-string*	Specifies the engine ID for the remote host.

	Command	Purpose
Step 3	**snmp-server user** *username groupname* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*]}	Configures an SNMP user to be associated with the remote host created in Step 2. **Note** You cannot configure a remote user for an address without first configuring the engine ID for the remote host. Otherwise, you receive an error message, and the command is not executed.
Step 4	**snmp-server group** *groupname* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]	Configures an SNMP group.
Step 5	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]	Specifies the recipient of an SNMP trap operation. • *host-addr*—Specifies the name or Internet address of the host (the targeted recipient). • (Optional) **informs**—Specifies SNMP informs to be sent to the host. • (Optional) **traps** (the default)—Specifies SNMP traps to be sent to the host. • (Optional) Specifies the SNMP **version** (**1**, **2c**, or **3**). SNMPv1 does not support informs. • (Optional) Version 3—Selects authentication level **auth, noauth**, or **priv**. **Note** The **priv** keyword is available only when the cryptographic software image is installed. • *community-string*—When **version 1** or **version 2c** is specified, enters the password-like community string sent with the notification operation. When **version 3** is specified, enter the SNMPv3 username. **Note** The @ symbol is used for delimiting the context information. Avoid using the @ symbol as part of the SNMP community string when configuring this command. • (Optional) *notification-type*—Specifies a notification type. Use the keywords listed in Table 36-4 on page 36-7. If no type is specified, all notifications are sent.
Step 6	**snmp-server enable traps** *notification-types*	Enables the switch to send traps or informs and specifies the type of notifications to be sent. For a list of notification types, see Table 36-4 on page 36-7, or enter **snmp-server enable traps ?** To enable multiple types of traps, you must enter a separate **snmp-server enable traps** command for each trap type. **Note** When you configure a trap by using the notification type **port-security**, configure the port security trap first, and then configure the port security trap rate: • **snmp-server enable traps port-security** • **snmp-server enable traps port-security trap-rate** *rate*

	Command	Purpose
Step 7	**snmp-server trap-source** *interface-id*	(Optional) Specifies the source interface, which provides the IP address for the trap message. This command also sets the source IP address for informs.
Step 8	**snmp-server queue-length** *length*	(Optional) Establishes the message queue length for each trap host. The range is 1 to 1000; the default is 10.
Step 9	**snmp-server trap-timeout** *seconds*	(Optional) Defines how often to resend trap messages. The range is 1 to 1000; the default is 30 seconds.
Step 10	**end**	Returns to privileged EXEC mode.

Setting the CPU Threshold Notification Types and Values

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**process cpu threshold type** {**total** | **process** | **interrupt**} **rising** *percentage* **interval** *seconds* [**falling** *fall-percentage* **interval** *seconds*]	Sets the CPU threshold notification types and values: • **total**—Sets the notification type to total CPU utilization. • **process**—Sets the notification type to CPU process utilization. • **interrupt**—Sets the notification type to CPU interrupt utilization. • **rising** *percentage*—Specifies the percentage (1 to 100) of CPU resources that, when exceeded for the configured interval, sends a CPU threshold notification. • **interval** *seconds*—Specifies the duration of the CPU threshold violation in seconds (5 to 86400) that, when met, sends a CPU threshold notification. • **falling** *fall-percentage*—Specifies the percentage (1 to 100) of CPU resources that, when usage falls below this level for the configured interval, sends a CPU threshold notification. This value must be equal to or less than the **rising** *percentage* value. If not specified, the **falling** *fall-percentage* value is the same as the **rising** *percentage* value.
Step 3	**end**	Returns to privileged EXEC mode.

Setting the Agent Contact and Location Information

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**snmp-server contact** *text*	Sets the system contact string.
Step 3	**snmp-server location** *text*	Sets the system location string.
Step 4	**end**	Returns to privileged EXEC mode.

Limiting TFTP Servers Used Through SNMP

	Command	Purpose
Step 1	**configure terminal**	Enters global configuration mode.
Step 2	**snmp-server tftp-server-list** *access-list-number*	Limits TFTP servers used for configuration file copies through SNMP to the servers in the access list. *access-list-number*—Enters an IP standard access list numbered from 1 to 99 and 1300 to 1999.
Step 3	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]	Creates a standard access list, repeating the command as many times as necessary. • *access-list-number*—Enters the access list number specified in Step 2. • **deny**—Denies access if the conditions are matched. The **permit** keyword permits access if the conditions are matched. • *source*—Enters the IP address of the TFTP servers that can access the switch. • (Optional) *source-wildcard*—Enters the wildcard bits, in dotted decimal notation, to be applied to the source. Place ones in the bit positions that you want to ignore. Recall that the access list is always terminated by an implicit deny statement for everything.
Step 4	**end**	Returns to privileged EXEC mode.

Monitoring and Maintaining SNMP

Command	Purpose
show snmp	Displays SNMP statistics.
show snmp engineID [**local** | **remote**]	Displays information on the local SNMP engine and all remote engines that have been configured on the device.
show snmp group	Displays information on each SNMP group on the network.
show snmp pending	Displays information on pending SNMP requests.
show snmp sessions	Displays information on the current SNMP sessions.
show snmp user	Displays information on each SNMP user name in the SNMP users table. **Note** You must use this command to display SNMPv3 configuration information for **auth** | **noauth** | **priv** mode. This information is not displayed in the **show running-config** output.

Configuration Examples for SNMP

Enabling SNMP Versions: Example

This example shows how to enable all versions of SNMP. The configuration permits any SNMP manager to access all objects with read-only permissions using the community string *public*. This configuration does not cause the switch to send any traps.

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

Permit SNMP Manager Access: Example

This example shows how to permit any SNMP manager to access all objects with read-only permission using the community string *public*. The switch also sends VTP traps to the hosts 192.180.1.111 and 192.180.1.33 using SNMPv1 and to the host 192.180.1.27 using SNMPv2C. The community string *public* is sent with the traps.

```
Switch(config)# snmp-server community public
Switch(config)# snmp-server enable traps vtp
Switch(config)# snmp-server host 192.180.1.27 version 2c public
Switch(config)# snmp-server host 192.180.1.111 version 1 public
Switch(config)# snmp-server host 192.180.1.33 public
```

Allow Read-Only Access: Example

This example shows how to allow read-only access for all objects to members of access list 4 that use the *comaccess* community string. No other SNMP managers have access to any objects. SNMP Authentication Failure traps are sent by SNMPv2C to the host *cisco.com* using the community string *public*.

```
Switch(config)# snmp-server community comaccess ro 4
Switch(config)# snmp-server enable traps snmp authentication
Switch(config)# snmp-server host cisco.com version 2c public
```

Configure SNMP Traps: Examples

This example shows how to send entity MIB traps to the host *cisco.com*. The community string is restricted. The first line enables the switch to send entity MIB traps in addition to any traps previously enabled. The second line specifies the destination of these traps and overwrites any previous **snmp-server host** commands for the host *cisco.com*.

```
Switch(config)# snmp-server enable traps entity
Switch(config)# snmp-server host cisco.com restricted entity
```

This example shows how to enable the switch to send all traps to the host *myhost.cisco.com* using the community string *public*:

```
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server host myhost.cisco.com public
```

Associating a User with a Remote Host: Example

This example shows how to associate a user with a remote host and to send **auth** (authNoPriv) authentication-level informs when the user enters global configuration mode:

```
Switch(config)# snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b
Switch(config)# snmp-server group authgroup v3 auth
Switch(config)# snmp-server user authuser authgroup remote 192.180.1.27 v3 auth md5
mypassword
Switch(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Switch(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Switch(config)# snmp-server enable traps
Switch(config)# snmp-server inform retries 0
```

Assigning a String to SNMP: Example

This example shows how to assign the string *comaccess* to SNMP, to allow read-only access, and to specify that IP access list 4 can use the community string to gain access to the switch SNMP agent:

```
Switch(config)# snmp-server community comaccess ro 4
```

Additional References

The following sections provide references related to switch administration:

Related Documents

Related Topic	Document Title
Cisco IE 2000 commands	*Cisco IE 2000 Switch Command Reference*, Release 15.0(1)EY
Cisco IOS basic commands	*Cisco IOS Configuration Fundamentals Command Reference*
Cisco IOS SNMP syntax and usage	*Cisco IOS Network Management Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature, and support for existing standards has not been modified by this feature.	—

MIBs

MIBs	MIBs Link
—	To locate and download MIBs using Cisco IOS XR software, use the Cisco MIB Locator found at the following URL and choose a platform under the Cisco Access Products menu: http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFCs

RFCs	Title
No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.	—

Technical Assistance

Description	Link
The Cisco Technical Support website contains thousands of pages of searchable technical content, including links to products, technologies, solutions, technical tips, and tools. Registered Cisco.com users can log in from this page to access even more content.	http://www.cisco.com/techsupport