Configuring System Message Logs

- Information About Configuring System Message Logs, on page 1
- How to Configure System Message Logs, on page 4
- Monitoring and Maintaining System Message Logs, on page 12
- Configuration Examples for System Message Logs, on page 12
- Additional References for System Message Logs, on page 13
- Feature History for System Message Logs, on page 13

Information About Configuring System Message Logs

System Messsage Logging

By default, a switch sends the output from system messages and **debug** privileged EXEC commands to a logging process. Member switches in a stack can trigger system messages. A member switch that generates a system message appends its hostname in the form of hostname-n, where n is a switch , and redirects the output to the logging process on the active switch . Though the active switch is a stack member, it does not append its hostname to system messages. The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration. The process also sends messages to the console.

When the logging process is disabled, messages are sent only to the console. The messages are sent as they are generated, so message and debug output are interspersed with prompts or output from other commands. Messages appear on the active consoles after the process that generated them has finished.

You can set the severity level of the messages to control the type of messages displayed on the consoles and each of the destinations. You can time-stamp log messages or set the syslog source address to enhance real-time debugging and management. For information on possible messages, see the system message guide for this release.

You can access logged system messages by using the switch command-line interface (CLI) or by saving them to a properly configured syslog server. The switch software saves syslog messages in an internal buffer on a standalone switch, and in the case of a switch stack, on the active switch . If a standalone switch or the active switch fails, the log is lost unless you had saved it to flash memory.

You can remotely monitor system messages by viewing the logs on a syslog server or by accessing the switch through Telnet, through the console port, or through the Ethernet management port. In a switch stack, all member switch consoles provide the same console output.

Note The syslog format is compatible with 4.3 BSD UNIX.

System Log Message Format

System log messages can contain up to 80 characters and a percent sign (%), which follows the optional sequence number or time-stamp information, if configured. Depending on the switch, messages appear in one of these formats:

- *seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)*
- *seq no:timestamp: %facility-severity-MNEMONIC:description*

The part of the message preceding the percent sign depends on the setting of these global configuration commands:

- **service sequence-numbers**
- **service timestamps log datetime**
- **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**]
- **service timestamps log uptime**

Table 1: System Log Message Elements

Element	Description
seq no:	Stamps log messages with a sequence number only if the **service sequence-numbers** global configuration command is configured.
timestamp formats: *mm/dd h h:mm:ss* or *hh:mm:ss* (short uptime) or *d h* (long uptime)	Date and time of the message or event. This information appears only if the **service timestamps log** [**datetime** | **log**] global configuration command is configured.
facility	The facility to which the message refers (for example, SNMP, SYS, and so forth).
severity	Single-digit code from 0 to 7 that is the severity of the message.
MNEMONIC	Text string that uniquely describes the message.
description	Text string containing detailed information about the event being reported.
hostname-n	Hostname of a stack member and its switch number in the stack. Though the active switch is a stack member, it does *not* append its hostname to system messages.

Default System Message Logging Settings

Table 2: Default System Message Logging Settings

Feature	Default Setting
System message logging to the console	Enabled.
Console severity	Debugging.
Logging file configuration	No filename specified.
Logging buffer size	102400 bytes.
Logging history size	1 message.
Time stamps	Disabled.
Synchronous logging	Disabled.
Logging server	Disabled.
Syslog server IP address	None configured.
Server facility	Local7
Server severity	Informational.

Syslog Message Limits

If you enabled syslog message traps to be sent to an SNMP network management station by using the **snmp-server enable trap** global configuration command, you can change the level of messages sent and stored in the switch history table. You also can change the number of messages that are stored in the history table.

Messages are stored in the history table because SNMP traps are not guaranteed to reach their destination. By default, one message of the level **warning** and numerically lower levels are stored in the history table even if syslog traps are not enabled.

When the history table is full (it contains the maximum number of message entries specified with the **logging history size** global configuration command), the oldest message entry is deleted from the table to allow the new message entry to be stored.

The history table lists the level keywords and severity level. For SNMP usage, the severity level values increase by 1. For example, *emergencies* equal 1, not 0, and *critical* equals 3, not 2.

How to Configure System Message Logs

Setting the Message Display Destination Device

If message logging is enabled, you can send messages to specific locations in addition to the console.

This task is optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**logging buffered** [*size*] **Example:** `Device(config)# logging buffered 8192`	Logs messages to an internal buffer on the switch. The range is 4096 to 2147483647 bytes. The default buffer size is 102400 bytes. If a standalone switchfails, the log file is lost unless you previously saved it to flash memory. See Step 4. **Note** Do not make the buffer size too large because the switch could run out of memory for other tasks. Use the **show memory** privileged EXEC command to view the free processor memory on the switch. However, this value is the maximum available, and the buffer size should *not* be set to this amount.
Step 3	**logging** *host* **Example:** `Device(config)# logging 125.1.1.100`	Logs messages to a UNIX syslog server host. *host* specifies the name or IP address of the host to be used as the syslog server. To build a list of syslog servers that receive logging messages, enter this command more than once.
Step 4	**logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*] **Example:** `Device(config)# logging file flash:log_msg.txt 40960 4096 3`	Stores log messages in a file in flash memory on a standalone switch. • *filename*—Enters the log message filename. • (Optional) **max-file-size** —Specifies the maximum logging file size. The range is

	Command or Action	Purpose
		4096 to 2147483647. The default is 4096 bytes. • (Optional) *min-file-size*—Specifies the minimum logging file size. The range is 1024 to 2147483647. The default is 2048 bytes. • (Optional) *severity-level-number* | *type*—Specifies either the logging severity level or the logging type. The severity range is 0 to 7.
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.
Step 6	**terminal monitor** **Example:** `Device# terminal monitor`	Logs messages to a nonconsole terminal during the current session. Terminal parameter-setting commands are set locally and do not remain in effect after the session has ended. You must perform this step for each session to see the debugging messages.

Synchronizing Log Messages

You can synchronize unsolicited messages and **debug** privileged EXEC command output with solicited device output and prompts for a specific console port line or virtual terminal line. You can identify the types of messages to be output asynchronously based on the level of severity. You can also configure the maximum number of buffers for storing asynchronous messages for the terminal after which messages are dropped.

When synchronous logging of unsolicited messages and **debug** command output is enabled, unsolicited device output appears on the console or printed after solicited device output appears or is printed. Unsolicited messages and **debug** command output appears on the console after the prompt for user input is returned. Therefore, unsolicited messages and **debug** command output are not interspersed with solicited device output and prompts. After the unsolicited messages appear, the console again displays the user prompt.

This task is optional.

Procedure

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

	Command or Action	Purpose
	`Device# configure terminal`	
Step 2	**line** [**console** | **vty**] *line-number* [*ending-line-number*] **Example:** `Device(config)# line console`	Specifies the line to be configured for synchronous logging of messages. • **console** —Specifies configurations that occur through the switch console port or the Ethernet management port. • **line vty** *line-number*—Specifies which vty lines are to have synchronous logging enabled. You use a vty connection for configurations that occur through a Telnet session. The range of line numbers is from 0 to 15. You can change the setting of all 16 vty lines at once by entering: line vty 0 15 You can also change the setting of the single vty line being used for your current connection. For example, to change the setting for vty line 2, enter: line vty 2 When you enter this command, the mode changes to line configuration.
Step 3	**logging synchronous** [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*] **Example:** `Device(config)# logging synchronous level 3 limit 1000`	Enables synchronous logging of messages. • (Optional) **level** *severity-level*—Specifies the message severity level. Messages with a severity level equal to or higher than this value are printed asynchronously. Low numbers mean greater severity and high numbers mean lesser severity. The default is 2. • (Optional) **level all**—Specifies that all messages are printed asynchronously regardless of the severity level. • (Optional) **limit** *number-of-buffers*—Specifies the number of buffers to be queued for the terminal after which new messages are dropped. The range is 0 to 2147483647. The default is 20.

	Command or Action	Purpose
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Disabling Message Logging

Message logging is enabled by default. It must be enabled to send messages to any destination other than the console. When enabled, log messages are sent to a logging process, which logs messages to designated locations asynchronously to the processes that generated the messages.

Disabling the logging process can slow down the switch because a process must wait until the messages are written to the console before continuing. When the logging process is disabled, messages appear on the console as soon as they are produced, often appearing in the middle of command output.

The **logging synchronous** global configuration command also affects the display of messages to the console. When this command is enabled, messages appear only after you press **Return**.

To reenable message logging after it has been disabled, use the **logging on** global configuration command.

This task is optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**no logging console** **Example:** `Device(config)# no logging console`	Disables message logging.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Enabling and Disabling Time Stamps on Log Messages

By default, log messages are not time-stamped.

This task is optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

Step 2	Use one of these commands:
	• **service timestamps log uptime**
	• **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]
	Example:
	`Device(config)# service timestamps log uptime`
	or
	`Device(config)# service timestamps log datetime`
		Enables log time stamps.
		• **log uptime**—Enables time stamps on log messages, showing the time since the system was rebooted.
		• **log datetime**—Enables time stamps on log messages. Depending on the options selected, the time stamp can include the date, time in milliseconds relative to the local time zone, and the time zone name.

Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Enabling and Disabling Sequence Numbers in Log Messages

If there is more than one log message with the same time stamp, you can display messages with sequence numbers to view these messages. By default, sequence numbers in log messages are not displayed.

This task is optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**service sequence-numbers** **Example:** `Device(config)# service sequence-numbers`	Enables sequence numbers.
Step 3	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Defining the Message Severity Level

Limit messages displayed to the selected device by specifying the severity level of the message.

This task is optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**logging console** *level* **Example:** `Device(config)# logging console 3`	Limits messages logged to the console. By default, the console receives debugging messages and numerically lower levels.
Step 3	**logging monitor** *level* **Example:** `Device(config)# logging monitor 3`	Limits messages logged to the terminal lines. By default, the terminal receives debugging messages and numerically lower levels.
Step 4	**logging trap** *level* **Example:** `Device(config)# logging trap 3`	Limits messages logged to the syslog servers. By default, syslog servers receive informational messages and numerically lower levels.

	Command or Action	Purpose
Step 5	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Limiting Syslog Messages Sent to the History Table and to SNMP

This task explains how to limit syslog messages that are sent to the history table and to SNMP.

This task is optional.

Procedure

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 2	**logging history** *level* **Example:** `Device(config)# logging history 3`	Changes the default level of syslog messages stored in the history file and sent to the SNMP server. By default, **warnings**, **errors**, **critical**, **alerts**, and **emergencies** messages are sent.
Step 3	**logging history size** *number* **Example:** `Device(config)# logging history size 200`	Specifies the number of syslog messages that can be stored in the history table. The default is to store one message. The range is 0 to 500 messages.
Step 4	**end** **Example:** `Device(config)# end`	Returns to privileged EXEC mode.

Logging Messages to a UNIX Syslog Daemon

This task is optional.

Note Some recent versions of UNIX syslog daemons no longer accept by default syslog packets from the network. If this is the case with your system, use the UNIX **man syslogd** command to decide what options must be added to or removed from the syslog command line to enable logging of remote syslog messages.

Before you begin

- Log in as root.
- Before you can send system log messages to a UNIX syslog server, you must configure the syslog daemon on a UNIX server.

Procedure

	Command or Action	Purpose
Step 1	Add a line to the file /etc/syslog.conf. **Example:** `local7.debug /usr/adm/logs/cisco.log`	• **local7**—Specifies the logging facility. • **debug**—Specifies the syslog level. The file must already exist, and the syslog daemon must have permission to write to it.

Step 2	Enter these commands at the UNIX shell prompt.
	Example:
	`$ touch /var/log/cisco.log`
	`$ chmod 666 /var/log/cisco.log`
		Creates the log file. The syslog daemon sends messages at this level or at a more severe level to this file.

Step 3	Make sure the syslog daemon reads the new changes. **Example:** `$ kill -HUP `cat /etc/syslog.pid``	For more information, see the **man syslog.conf** and **man syslogd** commands on your UNIX system.

Monitoring and Maintaining System Message Logs

Monitoring Configuration Archive Logs

Command	Purpose
show archive log config {**all** | **number** [*end-number*] | **user** *username* [**session** *number*] *number* [*end-number*] | **statistics**} [**provisioning**]	Displays the entire configuration log or the log for specified parameters.

Configuration Examples for System Message Logs

Example: Stacking System Message

This example shows a partial switch system message for an active switch and a stack member (hostname *Switch-2*):

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1, changed
state to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)

00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1, changed
state to down 2 (Switch-2)
```

Example: Switch System Message

This example shows a partial switch system message on a switch:

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state
 to down 2
*Mar  1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

```
*Mar  1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

Additional References for System Message Logs

Related Documents

Related Topic	Document Title
For complete syntax and usage information for the commands used in this chapter.	*Command Reference (Catalyst 9300 Series Switches)*

Feature History for System Message Logs

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Everest 16.5.1a	System Message Logs	A switch sends the output from system messages to a logging process. The logging process controls the distribution of logging messages to various destinations, such as the logging buffer, terminal lines, or a UNIX syslog server, depending on your configuration

Use Cisco Feature Navigator to find information about platform and software image support. To access Cisco Feature Navigator, go to http://www.cisco.com/go/cfn.