

System Message Logging

You can use system message logging to control the destination and to filter the severity level of messages that system processes generate. You can configure logging to terminal sessions, a log file, and syslog servers on remote systems.

- Overview, on page 1
- Guidelines and Limitations, on page 3
- Configure System Message Logging to Terminal Sessions, on page 5
- Configure Origin ID for Syslog Messages, on page 7
- Log System Messages to a File, on page 8
- Configure Module and Facility Messages Logging, on page 9
- Configure Logging Syslogs Compliant to RFC 5424, on page 12
- Configure Syslog Servers, on page 13
- Configure Secure Syslog Server, on page 15
- TCP Keepalive for Syslog Logging, on page 16
- Configure CA Certificate, on page 18
- Enroll CA Certificate, on page 19
- Configure Syslog Server on a UNIX or Linux System, on page 20
- Display and Clear Log Files, on page 21
- Configuration for System Message Logging, on page 22
- Related Documents, on page 24

Overview

System message logging is a mechanism that records system events and messages generated by the device, based on RFC 3164 .

- By default, the device outputs messages to terminal sessions and logs system messages to a log file.
- The device logs the most recent 100 messages of severity 0, 1, or 2 to the NVRAM log. Logging to NVRAM cannot be configured.
- You can configure which system messages are logged based on the facility that generated the message and its severity level.

Reference Information

For more information about the system message format and the messages that the device generates, see the Cisco NX-OS System Messages Reference .

The following table describes the severity levels used in system messages. When you configure the severity level, the system outputs messages at that level and lower.

Table 1: System Message Severity Levels

Level	Description
0 – emergency	System unusable
1 – alert	Immediate action needed
2 – critical	Critical condition
3 – error	Error condition
4 – warning	Warning condition
5 – notification	Normal but significant condition
6 – informational	Informational message only
7 – debugging	Appears during debugging only

Syslog Servers

The syslog servers run on remote systems that log system messages based on the syslog protocol. You can configure up to eight IPv4 or IPv6 syslog servers.

To support the same configuration of syslog servers on all switches in a fabric, you can use Cisco Fabric Services (CFS) to distribute the syslog server configuration.

Note When the device first initializes, messages are sent to syslog servers only after the network is initialized.

Secure Syslog Servers

Beginning with Cisco NX-OS Release 9.2(1), you can configure the syslog server with support for a secure TLS transport connectivity to remote logging servers. Additionally, you can enforce the NX-OS switches (client) identity via the mutual authentication configuration. For NX-OS switches, this feature supports TLSv1.1 and TLSv1.2.

The Secure syslog server feature uses the TCP/TLS transport and security protocols to provide device authentication and encryption. This feature enables a Cisco NX-OS device (acting as a client) to make a secure, encrypted outbound connection to remote syslog servers (acting as a server) supporting secure connectivity for logging. With authentication and encryption, this feature allows for a secure communication over an insecure network.

Default Settings

The following table lists the default settings for the system message logging parameters.

Table 2: Default System Message Logging Parameters

Parameters	Default
Console logging	Enabled at severity level 2
Monitor logging	Enabled at severity level 5
Log file logging	Enabled to log messages at severity level 5
Module logging	Enabled at severity level 5
Facility logging	Enabled
Time-stamp units	Seconds
Syslog server logging	Disabled
Syslog server configuration distribution	Disabled

Guidelines and Limitations

System message logging has the following configuration guidelines and limitations:

General Guidelines

- System messages are logged to the console and the log file by default.
- Any system messages that are printed before the syslog server is reachable (such as supervisor active or online messages) cannot be sent to the syslog server.
- Cisco recommends maintaining the logging levels for all processes at default. Increasing the levels to higher values can result in seeing syslog messages that are not intended for customers, can generate false alarms, and are generally supposed to be used for short-term troubleshooting purposes by TAC. Cisco does not provide support for syslog messages at levels above default.
- Due to limitations in Syslog, securePOAP pem file name characters length is limited to 230 characters, though secure POAP supports 256 characters length for a pem file name.
- The maximum length of syslog messages can be 324 characters. If a message exceeds this limit, NX-OS truncates the displayed message. Truncation does not indicate that the underlying operation failed.
- For the secure syslog server(s) to be reachable over an in-band (nonmanagement) interface, the CoPP profile may need tweaks. Especially when multiple logging servers are configured and when many syslogs are generated in a short time (such as, boot up and config application).
- This guideline applies to the user-defined persistent logging file:

  The syslog command, **logging logfile** , allows the configuration of the logfile both in persistent (/logflash/log) and non-persistent locations (/log).

The default logfile is named “messages” and this file, along with backup files (if present) messages.1, messages.2, messages.3, messages.4 cannot be deleted, even by the **delete /log/** or **delete logflash:/log/** commands.

There is a provision to configure custom-named logfiles (**logging logfile** *file-name severity*), however this custom-named file can be deleted by the delete operation. If this occurs, syslog logging does not function.

For example, the custom-named logfile is configured and the same file gets deleted via delete operation. Because this is an intentional delete operation, in order to log the syslog messages on the custom logfiles, you must reconfigure the custom logfile using command **logging logfile** *file-name severity* . Until this configuration is performed, the syslog logging cannot occur.

- Generally, the syslogs display the local time zone. However, few components such as NGINX display the logs in UTC time zone.

Release Specific Guidelines

Beginning with Cisco NX-OS Release 9.2(1)

- You can configure the syslog server with support for a secure TLS transport connectivity to remote logging servers. This feature supports TLS v1.1 and TLS v1.2.

Beginning with Cisco NX-OS Release 10.2(4)M

- TLS v1.3 is supported for syslog on Cisco Nexus 9000 series platform switches.

Beginning with Cisco NX-OS Release 10.4(3)F

- TLS v1.2 and TLS v1.3 is supported for syslog on Cisco Nexus 9000 Series platform switches. TLS v1.1 and TLS v1.0 support for syslog is deprecated.

Beginning with Cisco NX-OS Release 10.3(4a)M

- The existing **logging rfc-strict 5424** command (optional) that enables the syslog protocol RFC 5424 is enhanced by adding a new keyword (**full**) as follows:

  logging rfc-strict 5424 full

  The addition of this keyword ensures complete compliance with the RFC 5424 standard for Syslog Protocol. However, if the values are not available for the `[APP-NAME] [PROCID] [MSG-ID] [STRUCTRED-DATA]` fields, then the nil value is indicated by a dash (-).

Beginning with Cisco NX-OS Release 10.5(3)

- The existing **logging rfc-strict 5424** command (optional) that enables the syslog protocol RFC 5424 is enhanced by adding a new keyword (**utc**) as follows:

  logging rfc-strict 5424 utc

  The addition of this keyword enables the RFC 5424 standard for Syslog Protocol with UTC time format.

  You can also ensure complete compliance with the RFC 5424 standard for Syslog Protocol in UTC time format with the following command: **logging rfc-strict 5424 utc full** .

Starting with Cisco NX-OS Release 10.6(3), console logging is handled asynchronously so that syslogd continues processing and delivering log messages without waiting on console output. This helps ensure that slow console output does not slow down other logging destinations or system operations.

- Use the existing logging drop statistics commands. The output is enhanced to include console-specific drop/error details.

```
show logging dropcount
```

```
clear logging dropcount
```

- Console logging uses a fixed internal buffer (300 messages). If the buffer becomes full, additional console messages can be dropped instead of slowing down syslogd.
- To maintain responsiveness, syslogd does not retry the same console message after a console write failure.

Configure System Message Logging to Terminal Sessions

You can configure the device to log messages by their severity level to console, Telnet, and SSH sessions.

By default, logging is enabled for terminal sessions.



Note The current critical (default) logging level is maintained if the console baud speed is 9600 baud (default). All attempts to change the console logging level will generate an error message. To increase the logging level (above critical), you must change the console baud speed to 38400 baud.



Note Be aware that the Cisco NX-OS commands for this feature might differ from those commands used in Cisco IOS.

Procedure

Step 1 Enable the device to log messages to the console using the command **terminal monitor**

Example:

```
switch# terminal monitor
```

Step 2 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 3 Configure the device to log messages to the console session based on a specified severity level or higher using the command [**no**] **logging console** [*severity-level*]

Example:

```
switch(config)# logging console 3
```

A lower number indicates a higher severity level. Severity levels range from 0 to 7:

- 0 – emergency
- 1 – alert
- 2 – critical
- 3 – error
- 4 – warning
- 5 – notification
- 6 – informational
- 7 – debugging

If the severity level is not specified, the default of 2 is used. The **no** option disables the device's ability to log messages to the console.

Step 4 Display the console logging configuration using the command **show logging console**

Example:

```
switch(config)# show logging console
```

This is an optional step.

Step 5 Enable the device to log messages to the monitor based on a specified severity level or higher using the command [**no**] **logging monitor** [*severity-level*]

Example:

```
switch(config)# logging monitor 3
```

A lower number indicates a higher severity level. Severity levels range from 0 to 7:

- 0 – emergency
- 1 – alert
- 2 – critical
- 3 – error
- 4 – warning
- 5 – notification
- 6 – informational
- 7 – debugging

The configuration applies to Telnet and SSH sessions.

If the severity level is not specified, the default of 2 is used. The **no** option disables the device's ability to log messages to the Telnet and SSH sessions.

Step 6 (Optional) Display the monitor logging configuration using the command **show logging monitor**

Example:

```
switch(config)# show logging monitor
```

This is an optional step.

Step 7 Add the description for physical Ethernet interfaces and subinterfaces in the system message log using the command [**no**] **logging message interface type ethernet description**

Example:

```
switch(config)# logging message interface type ethernet description
```

The description is the same description that was configured on the interface.

The **no** option disables the printing of the interface description in the system message log for physical Ethernet interfaces.

Step 8 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure Origin ID for Syslog Messages

You can configure Cisco NX-OS to append the hostname, an IP address, or a text string to syslog messages that are sent to remote syslog servers.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Specify the hostname, IP address, or text string to be appended to syslog messages that are sent to remote syslog servers using the command **logging origin-id** { **hostname** | **ip** *ip-address* | **string** *text-string* }

Example:

```
switch(config)# logging origin-id string n9k-switch-abc
```

Step 3 Display the configured hostname, IP address, or text string that is appended to syslog messages that are sent to remote syslog servers using the command **show logging origin-id**

Example:

```
switch(config)# show logging origin-id
 Logging origin_id : enabled (string: n9k-switch-abc)
```

This is an optional step.

Step 4 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Log System Messages to a File

You can configure the device to log system messages to a file. By default, system messages are logged to the file /logflash/log/ *logfilename* .

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
  switch(config)#
```

Step 2 Configure the nonpersistent or persistent log file parameters using the command [**no**] **logging logfile** *logfile-name severity-level* [**persistent threshold** *percent* | **size** *bytes*]

Example:

```
switch(config)# logging logfile my_log 6

switch(config)# logging logfile my_log 6 persistent threshold 90
```

logfile-name : Configures the name of the log file that is used to store system messages. Default filename is "message".

severity-level : Configures the minimum severity level to log. A lower number indicates a higher severity level. Default is 5. Range is from 0 through 7:

- 0 – emergency
- 1 – alert
- 2 – critical
- 3 – error
- 4 – warning
- 5 – notification
- 6 – informational
- 7 – debugging

persistent threshold *percent* : Optionally configure the threshold percentage for the persistent log file. Range is from 0 through 99.

Note
Setting **persistent threshold** to 0 (zero) disables the persistent threshold feature and generates no threshold syslogs.

percent configures the percent threshold size of the persistent file. Once the threshold size is reached, an alert notification message is logged. On reaching 100% utilization of the persistent log file, the system sends another syslog message notification. The system then creates a backup file of the existing log file and starts writing into a new log file with the configured threshold percentage applied. In total, the last five backup files are present at most. After five files, the system deletes files based on the oldest modified.

Note
Persistent logging is a system-enabled feature. Log files are located here: /logflash/log/[filename].

Outputs of the following show commands support the persistent log file feature:

- **show logging info**
- **show logging**

The outputs include the following persistent logging information:

```
Logging logflash: enabled (Severity: notifications)(threshold percentage: 99)
 Logging logfile: enabled
 Name - messages: Severity - notifications Size - 4194304
```

size *bytes* : Optionally specify maximum file size. Range is from 4096 through 4194304 bytes.

Step 3 Log interface events using the command **logging event** { **link-status** | **trunk-status** } { **enable** | **default** }

Example:

```
switch(config)# logging event link-status default
```

- **link-status** —Logs all UP/DOWN and CHANGE messages.
- **trunk-status** —Logs all TRUNK status messages.
- **enable** —Specifies to enable logging to override the port level configuration.
- **default** —Specifies that the default logging configuration is used by interfaces that are not explicitly configured.

Step 4 Display the logging configuration using the command **show logging info**

Example:

```
switch(config)# show logging info
```

This is an optional step.

Step 5 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure Module and Facility Messages Logging

You can configure the severity level and time-stamp units of messages logged by modules and facilities.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
 switch(config)#
```

Step 2 Enable module log messages that have the specified severity level or higher using the command [**no**] **logging module** [*severity-level*]

Example:

```
switch(config)# logging module 3
```

Severity levels range from 0 to 7:

- 0 – emergency
- 1 – alert
- 2 – critical
- 3 – error
- 4 – warning
- 5 – notification
- 6 – informational
- 7 – debugging

If the severity level is not specified, the default of 5 is used. The **no** option disables module log messages.

Step 3 Display the module logging configuration using the command **show logging module**

Example:

```
switch(config)# show logging module
```

This is an optional step.

Step 4 Enable logging messages from the specified facility that have the specified severity level or higher using the command [**no**] **logging level** *facility severity-level*

Example:

```
switch(config)# logging level aaa 2
```

Severity levels range from 0 to 7:

- 0 – emergency
- 1 – alert
- 2 – critical
- 3 – error

- 4 – warning
- 5 – notification
- 6 – informational
- 7 – debugging

To apply the same severity level to all facilities, use the **all** facility. For defaults, see the **show logging level** command.

The **no** option resets the logging severity level for the specified facility to its default level. If you do not specify a facility and severity level, the device resets all facilities to their default levels.

Step 5 Display the logging level configuration and the system default level by facility using the command **show logging level** [*facility*]

Example:

```
switch(config)# show logging level aaa
```

This is an optional step. If you do not specify a facility, the device displays levels for all facilities.

Note
In running configurations, the logging level for authpriv is displayed as authpri in releases earlier than 10.4(3)F and as authpriv from release 10.4(3)F.

Step 6 Enable logging of the Ethernet Port Manager link-up/link-down syslog messages at level 3 using the command [**no**] **logging level** *ethpm*

Example:

```
     switch(config)# logging level ethpm ?
     <0-7>      0-emerg;1-alert;2-crit;3-err;4-warn;5-notif;6-inform;7-debug
link-down  Configure logging level for link down syslog messages
link-up    Configure logging level for link up syslog messages

     switch(config)#logging level ethpm link-down ?
     error  ERRORS
notif  NOTICE
     (config)# logging level ethpm link-down error ?

     <CR>
     (config)# logging level ethpm link-down notif ?
     <CR>
     switch(config)#logging level ethpm link-up ?
     error  ERRORS
notif  NOTICE
     (config)# logging level ethpm link-up error ?

     <CR>
     (config)# logging level ethpm link-up notif ?
     <CR>
```

Use the **no** option to use the default logging level for Ethernet Port Manager syslog messages.

Step 7 Set the logging time-stamp units using the command [**no**] **logging timestamp** { **microseconds** | **milliseconds** | **seconds** }

Example:

```
switch(config)# logging timestamp milliseconds
```

By default, the units are seconds.

Note
This command applies to logs that are kept in the switch. It does not apply to the external logging server.

Step 8 Display the logging time-stamp units configured using the command **show logging timestamp**

Example:

```
switch(config)# show logging timestamp
```

This is an optional step.

Step 9 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure Logging Syslogs Compliant to RFC 5424

The command can be modified in the following ways :

- [*no*] **logging rfc-strict 5424**
- **show logging rfc-strict 5424**

Procedure

Step 1 `switch(config)#` [*no*] logging rfc-strict 5424

(optional) Negate a command or set its defaults

Step 2 `switch(config)#` **logging rfc-strict 5424**

Modify message logging facilities and set RFC to which messages should be compliant.

Step 3 `switch(config)#` show logging rfc-strict 5424

Displays the syslogs which will be compliant to RFC 5424

Configure Syslog Servers

Note Cisco recommends that you configure the syslog server to use the management virtual routing and forwarding (VRF) instance. For more information on VRFs, see Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide.

You can configure up to eight syslog servers that reference remote systems where you want to log system messages.

Note Until Cisco NX-OS Release 10.3(2)F, when the user input certain default values, the running-config of logging server commands showed those default values randomly or inconsistently. However, beginning with Cisco NX-OS Release 10.3(2)F, the running config consistently shows only the non-default values.

For example, in earlier releases, for a certain user input, if the running-config showed `logging server 1.1.1.1 port 514 facility local7 use-vrf default` values, from Cisco NX-OS Release 10.3(2)F onwards, for the same input, the running-config shows only `logging server 1.1.1.1` value. Notice that the default value such as the default port, default facility (local7), and the default VRF are not shown in the running-config.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
  switch(config)#
```

Step 2 Configure a syslog server at the specified hostname, IPv4, or IPv6 address using the command [**no**] **logging server** *host* [*severity-level* [**use-vrf** *vrf-name*]]

Example:

```
switch(config)# logging server 192.0.2.253
```

Example:

```
switch(config)# logging server 2001::3 5 use-vrf red
```

You can specify logging of messages to a particular syslog server in a VRF by using the **use-vrf** keyword. The **use-vrf** *vrf-name* keyword identifies the default or management values for the VRF name. The default VRF is marked as 'default'. However, the **show-running** command will not list the default VRF. Severity levels range from 0 to 7:

- 0 – emergency
- 1 – alert

- 2 – critical
- 3 – error
- 4 – warning
- 5 – notification
- 6 – informational
- 7 – debugging

The default outgoing facility is local7.

The **no** option removes the logging server for the specified host.

The first example forwards all messages on facility local 7. The second example forwards messages with severity level 5 or lower to the specified IPv6 address in VRF red.

Note
After configuring this command, any one of the following server status is displayed:

- **Configured** – Configuration is successful.
- **No errors found** - If the syslog is transmitted to the remote syslog server successfully, this status is displayed.
- **Temporarily unreachable** - If there is a problem with transmission, this status is displayed. However, internally, the system probes the problem with transmission. After a while, when the issue is resolved, the status changes to No errors found.

Step 3 Enable a source interface for the remote syslog server using the command **logging source-interface loopback** *virtual-interface*

Example:

```
switch(config)# logging source-interface loopback 5
```

The range for the *virtual-interface* argument is from 0 to 1023.

Step 4 Display the syslog server configuration using the command **show logging server**

Example:

```
switch(config)# show logging server
```

This is an optional step.

Step 5 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Configure Secure Syslog Server

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
                        switch(config)#
```

Step 2 Configure a syslog server at the specified hostname or IPv4 or IPv6 address using the command [**no**] **logging server** *host* [*severity-level* [**port** *port-number*][**secure** [**trustpoint client-identity** *trustpoint-name*]][**use-vrf** *vrf-name*]]

Example:

One-Way TLS Authentication (Server Authentication)

To enable one-way TLS authentication, use the secure keyword. The switch verifies the identity of the syslog server using the CA certificates installed on the switch.

```
switch(config)# logging server 192.0.2.253 secure
```

Example:

Mutual TLS Authentication (Two-Way Authentication)

To enable mutual TLS authentication, use the secure trustpoint client-identity trustpoint-name option. The switch and syslog server validate each other's identity. The switch presents the certificate configured in the specified trustpoint to the server.

```
switch(config)# logging server 2001::3 5 secure trustpoint client-identity myCA use-vrf red
```

Note
The default destination port for secure TLS connections is 6514. Configure and authenticate the required CA trustpoints before enabling secure logging.

Optionally, you can enforce a mutual authentication by installing the client identity certificate that is signed by any CA and using the trustpoint client-identity option.

The default destination port for a secure TLS connection is 6514.

Step 3 Enable a source interface for the remote syslog server using the command **logging source-interface** *interface name*

Example:

```
switch(config)# logging source-interface lo0
```

This is an optional step.

Step 4 Display the syslog server configuration using the command **show logging server**

Example:

```
switch(config)# show logging server
```

This is an optional step. If the secure option is configured, the output will have an entry with the transport information. By default, the transport is UDP if the secure option is not configured.

Step 5 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

In Cisco NX-OS Release 9.3(8), when the OCSP responder is down, or if there are OCSP signing issues, SSL connection fails as OCSP works in a strict mode. Hence, beginning with Cisco NX-OS Release 9.3(9), the following new command is introduced to allow you to enable or disable strict-mode.

[**no**] **logging secure ocsp strict**



Note By default, the strict-mode is enabled. Use the no form of the command to enable the non-strict-mode.

TCP Keepalive for Syslog Logging

The TCP Keepalive for Syslog Logging feature improves the reliability of syslog message delivery by maintaining active TCP connections to remote syslog servers, even during periods of inactivity.

- Allows administrators to configure a TCP keepalive idle timeout for syslog over TCP connections.
- Automatically sends keepalive probes during idle periods to prevent connection drops caused by network middleboxes (such as firewalls and load balancers).
- Detects and recovers from dropped TCP connections to ensure continuous syslog logging.

This feature is essential for environments where syslog messages are sent to remote servers over TCP and network devices may enforce idle timeout policies.

- Prevents silent loss of syslog messages by keeping the TCP connection alive, ensuring important audit and security logs are not missed.
- Reduces the need for manual intervention by automatically re-establishing connections if they are dropped due to inactivity.
- Provides flexibility for administrators to tune the idle timeout value to suit their network environment, improving integration with existing security policies and middleboxes.

Guidelines and Limitations for TCP Keepalive

Follow these guidelines and limitations when configuring TCP keepalive for syslog logging.

- The configurable idle timeout range is 300 to 7200 seconds. The default idle timeout is 7200 seconds.
- Once the idle timeout is configured, it can only be reverted to the default value (7200 seconds) by using the `no logging tcp keepalive idle-timeout` command. Restarting the device does not reset the idle timeout setting.

- Keepalive probe interval and probe count are fixed at 75 seconds and 9 probes.
- This feature applies only to syslog servers configured with TCP transport.
- Disabling or misconfiguring keepalive can result in undetected loss of syslog messages.
- Feature configuration changes take effect only for new TCP syslog connections. Existing connections may not inherit updated keepalive settings until they are re-established.
- The idle timeout setting is global and applies to all active syslog servers on the switch. You cannot set different idle timeout values for individual syslog servers.

Configure TCP Keepalive for Syslog Logging

Configure TCP keepalive when syslog is sent over TCP to a remote server and there is a risk of idle connections being dropped.

TCP keepalive is needed when network segments (such as firewalls or load balancers) enforce idle timeouts on TCP connections, which could disrupt syslog logging.

Before you begin

Ensure the following before configuring TCP keepalive for syslog logging:

- Syslog servers are configured to use TCP transport.
- Know the desired idle timeout value for your environment.

Follow these steps to configure TCP keepalive for syslog logging.

Procedure

Step 1 Use the `configure terminal command` to enter global configuration mode.

Example:

```
switch# configure terminal
switch(config)#
```

Step 2 Use the `logging tcp keepalive idle-timeout` **`idle-timeout value`** command to configure the TCP keepalive idle timeout for syslog logging.

Example:

```
switch(config)# logging tcp keepalive idle-timeout 300
```

Replace *idle-timeout value* with the desired idle timeout in seconds. The default value is 7200 seconds.

Note
Only integer values within the range of 300 to 7200 are accepted for the idle timeout. Decimal values are not supported.

Step 3 (Optional) Use the `no logging tcp keepalive idle-timeout` command to revert to the default idle timeout.

Example:

```
switch(config)# no logging tcp keepalive idle-timeout
```

This restores the default idle timeout value (7200 seconds).

Step 4 Use the exit command to leave configuration mode.

Example:

```
switch(config)# exit
```

Step 5 Use the `show running-config | grep "idle"` command to verify the configured TCP keepalive idle timeout for syslog logging.

Example:

```
switch(config)#show running-config | grep "idle"
logging tcp idle-timeout 300
```

If the idle timeout is set to a non-default value, it will be displayed in the output.

Step 6 Use the `show running-config all | grep "idle"` command to check the TCP keepalive idle timeout, including the default value.

Example:

```
switch# show running-config all | grep "idle"
logging tcp idle-timeout 7200
```

If you do not configure the idle timeout, the `show running-config | grep "idle"` command does not show any output. To verify the idle timeout value, use `show running-config all | grep "idle"`. This command shows the default value (7200 seconds) if you have not set a different value.

The switch now maintains active TCP connections for syslog logging according to the configured keepalive idle timeout, reducing the risk of log loss due to idle timeouts.

Configure CA Certificate

For the secure syslog feature support, the remote servers must be authenticated via a trustpoint configuration.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
                        switch(config)#
```

Step 2 Configure a trustpoint using the command [**no**] **crypto ca trustpoint** *trustpoint-name*

Example:

```
switch(config)# crypto ca trustpoint winca
                        switch(config-trustpoint)#
```

Note
You must configure the ip domain-name before the trustpoint configuration.

Step 3 Configure a CA certificate for the trustpoint using the command **crypto ca authenticate** *trustpoint-name*

Example:

```
switch(config-trustpoint)# crypto ca authenticate winca
```

Step 4 Display the configured certificate/chain and the associated trustpoint using the command **show crypto ca certificate**

Example:

```
switch(config)# show crypto ca certificates
```

This is an optional step.

Step 5 Copy the running configuration to the startup configuration so that the trustpoint is persistent across the reload of the device using the command **copy running-config startup-config**

Example:

```
switch(config)# copy running-config startup-config
```

This is an optional step.

Enroll CA Certificate

For mutual authentication, where the remote server wants the NX-OS switch (the client) to identify, that the peer authentication is mandatory, this is an additional configuration to enroll the certificate on the switch.

Procedure

Step 1 Enter global configuration mode using the command **configure terminal**

Example:

```
switch# configure terminal
                        switch(config)#
```

Step 2 Configure an RSA key pair using the command **crypto key generate rsa label** *key name* **exportable modules 2048**

Example:

```
switch(config-trustpoint)# crypto key generate rsa label myKey exportable modulus 2048
```

By default, the Cisco NX-OS software generates an RSA key using 1024 bits. Repeat step 2.

Step 3 Associate the keypair generated to the trustpoint CA using the command **rsakeypair** *key-name*

Example:

```
switch(config-trustpoint)# rsakeypair myKey
```

Step 4 Configure a CA certificate for the trustpoint using the command **crypto ca trustpoint** *trustpoint-name*

Example:

```
switch(config)# crypto ca authenticate myCA
```

Step 5 Generate an identity certificate of the switch to enroll it to a CA using the command [**no**] **crypto ca enroll** *trustpoint-name*

Example:

```
switch(config)# crypto ca enroll myCA
```

Step 6 Import the identity certificate signed by the CA to the switch using the command **crypto ca import** *trustpoint-name* **certificate**

Example:

```
switch(config-trustpoint)# crypto ca import myCA certificate
```

Step 7 Display the configured certificate or chain and the associated trustpoint using the command **show crypto ca certificates**

Example:

```
switch# show crypto ca certificates
```

This is an optional step.

Step 8 Copy the running configuration to the startup configuration using the command **copy running-config startup-config**

Example:

```
switch# copy running-config startup-config
```

Configure Syslog Server on a UNIX or Linux System

You can configure a syslog server on a UNIX or Linux system by adding the following line to the /etc/syslog.conf file:

```
    facility.level
<five tab characters>
    action
```

The following table describes the syslog fields that you can configure.

Table 3: Syslog fields in syslog.conf

Field	Description
Facility	Creator of the message, which can be auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, syslog, user, local0 through local7, or an asterisk (*) for all. These facility designators allow you to control the destination of messages based on their origin. **Note** Check your configuration before using a local facility.
Level	Minimum severity level at which messages are logged, which can be debug, info, notice, warning, err, crit, alert, emerg, or an asterisk (*) for all. You can use none to disable a facility.

Field	Description
Action	Destination for messages, which can be a filename, a hostname preceded by the at sign (@), a comma-separated list of users, or an asterisk (*) for all logged-in users.

Procedure

Step 1 Log debug messages with the local7 facility in the file /var/log/myfile.log by adding the following line to the /etc/syslog.conf file

Example:

```
debug.local7 var/log/myfile.log
```

Step 2 Create the log file by entering these commands at the shell prompt

Example:

```
$ touch /var/log/myfile.log
 $ chmod 666 /var/log/myfile.log
```

Step 3 Make sure the system message logging daemon reads the new changes by checking myfile.log after entering this command

Example:

```
$ kill -HUP ~cat /etc/syslog.pid~
```

Display and Clear Log Files

You can display or clear messages in the log file and the NVRAM.

Procedure

Step 1 Required: Display the last number of lines in the logging file using the command **show logging last** *number-lines*

Example:

```
switch# show logging last 40
```

You can specify from 1 to 9999 for the last number of lines.

Step 2 Display the messages in the log file that have occurred within the duration entered using the command **show logging logfile duration** *hh:mm:ss*

Example:

```
switch# show logging logfile duration 15:10:0
```

Step 3 Display the sequence number of the last message in the log file using the command **show logging logfile last-index**

Example:

```
switch# show logging logfile last-index
```

Step 4 Display the messages in the log file that have a timestamp within the span entered using the command **show logging logfile** [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]

Example:

```
switch# show logging logfile start-time 2013 oct 1 15:10:0
```

If you do not enter an end time, the current time is used. You enter three characters for the month time field and digits for the year and day time fields.

Step 5 Display messages occurring within a range of sequence numbers using the command **show logging logfile** [**start-seqn** *number*] [**end-seqn** *number*]

Example:

```
switch# show logging logfile start-seqn 100 end-seqn 400
```

If you do not include an end sequence number, the system displays messages from the start number to the last message in the log file.

Step 6 Display the messages in the NVRAM using the command **show logging nvram** [**last** *number-lines*]

Example:

```
switch# show logging nvram last 10
```

To limit the number of lines displayed, you can enter the last number of lines to display. You can specify from 1 to 100 for the last number of lines.

Step 7 Clear the contents of the log file using the command **clear logging logfile** [**persistent**]

Example:

```
switch# clear logging logfile
```

persistent : Clears the contents of the log file from the persistent location.

Step 8 Clear the logged messages in NVRAM using the command **clear logging nvram**

Example:

```
switch# clear logging nvram
```

Configuration for System Message Logging

System Message Logging Configuration Example

System message logging configuration enables you to control where and how system messages are logged on the device.

This example shows how to configure system message logging:

```
configure terminal
    logging console 3
    logging monitor 3
```

```
logging logfile my_log 6
logging module 3
logging level aaa 2
logging timestamp milliseconds
logging server 172.28.254.253
logging server 172.28.254.254 5 facility local3
copy running-config startup-config
```

Repeated System Logging Messages

System processes generate logging messages. Depending on the filters used to control which severity levels are generated, a large number of messages can be produced with many of them being repeated.

To make it easier to develop scripts to manage the volume of logging messages, and to eliminate repeated messages from "flooding" the output of the **show logging log** command, the following method of logging repeated messages is used.

In the old method, when the same message was repeated, the default was to state the number of times it reoccurred in the message:

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5
2019 Mar 11 13:43:15 Cisco-customer last message repeated 242 times
```

The new method simply appends the repeat count to the end of the repeated message:

```
2019 Mar 11 13:42:44 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5

2019 Mar 11 13:43:15 Cisco-customer %PTP-2-PTP_INCORRECT_PACKET_ON_SLAVE:
Incorrect delay response packet received on slave interface Eth1/48 by
2c:5a:0f:ff:fe:51:e9:9f. Source Port Identity is 08:00:11:ff:fe:22:3e:4e. Requesting Port
Identity is 00:1c:73:ff:ff:ee:f6:e5 (message repeated 242 times)
```

Verify System Message Logging Configuration

To display system message logging configuration information, perform one of the following tasks:

Command	Purpose
show logging console	Displays the console logging configuration.
show logging info	Displays the logging configuration.
show logging last *number-lines*	Displays the last number of lines of the log file.
show logging level [*facility*]	Displays the facility logging severity level configuration.
show logging logfile duration *hh:mm:ss*	Displays the messages in the log file that have occurred within the duration entered.
show logging logfile last-index	Displays the sequence number of the last message in the log file.

Command	Purpose
show logging logfile [**start-time** *yyyy mmm dd hh:mm:ss*] [**end-time** *yyyy mmm dd hh:mm:ss*]	Displays the messages in the log file based on a start and end date/time.
show logging logfile [**start-seqn** *number*] [**end-seqn** *number*]	Displays messages occurring within a range of sequence numbers. If you do not include an end sequence number, the system displays messages from the start number to the last message in the log file.
show logging module	Displays the module logging configuration.
show logging monitor	Displays the monitor logging configuration.
show logging nvram [**last** *number-lines*]	Displays the messages in the NVRAM log.
show logging server	Displays the syslog server configuration.
show logging timestamp	Displays the logging time-stamp units configuration.

Related Documents

Related Topic	Document Title
Clock manager	Cisco Nexus 9000 Series NX-OS Fundamentals Configuration Guide