

Implementing System Logging

This module describes the tasks you need to implement logging services on the router.

The Cisco IOS XR Software provides basic logging services. Logging services provide a means to gather logging information for monitoring and troubleshooting, to select the type of logging information captured, and to specify the destinations of captured system logging (syslog) messages.

- Implementing System Logging , on page 1

Implementing System Logging

System Logging (Syslog) is the standard application used for sending system log messages. Log messages indicates the health of the device and point to any encountered problems or simplify notification messages according to the severity level. The IOS XR router sends its syslog messages to a syslog process. By default, syslog messages will be sent to the console terminal. But, syslog messages can be send to destinations other than the console such as the logging buffer, syslog servers, and terminal lines.

Syslog Message Format

By default, the general format of syslog messages generated by the syslog process on the Cisco IOS XR software is as follows:

```
node-id : timestamp : process-name [pid] : % message category -group -severity -message
-code : message-text
```

The following table describes the general format of syslog messages on Cisco IOS XR software.

Table 1: Format of Syslog Messages

Field	Description
node-id	Node from which the syslog message originated.
timestamp	Time stamp in the month day HH:MM:SS format, indicating when the message was generated. **Note** The time-stamp format can be modified using the **service timestamps** command.
process-name	Process that generated the syslog message.

Field	Description
size	Process ID (pid) of the process that generated the syslog message.
[pid]	Message category, group name, severity, and message code associated with the syslog message.
message-text	Text string describing the syslog message.

Syslog Message Severity Levels

In the case of logging destinations such as console terminal, syslog servers and terminal lines, you can limit the number of messages sent to a logging destination by specifying the severity level of syslog messages. However, for the logging buffer destination, syslog messages of all severity will be sent to it irrespective of the specified severity level. In this case, the severity level only limits the syslog messages displayed in the output of the command **show logging**, at or below specified value. The following table lists the severity level keywords that can be supplied for the severity argument and the corresponding UNIX syslog definitions in order from the most severe level to the least severe level.

Table 2: Syslog Message Severity Levels

Severity Keyword	Level	Description
emergencies	0	System unusable
alert	1	Immediate action needed
critical	2	Critical conditions
errors	3	Error conditions
warnings	4	Warning conditions
notifications	5	Normal but significant condition
informational	6	Informational messages only
debugging	7	Debugging messages

Prerequisites for Configuring System Logging

These prerequisites are required to configure the logging of system messages in your network operating center (NOC):

- You must be in a user group associated with a task group that includes the proper task IDs. The command reference guides include the task IDs required for each command. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.
- You must have connectivity with syslog servers to configure syslog server hosts as the recipients for syslog messages.

Third-party applications logging on remote server

Table 3: Feature History Table

Feature Name	Release Information	Feature Description
Third-party applications logging on remote server	Release 24.4.1	This feature enables third-party applications to forward syslog messages to a remote server for handling a high rate of system logging.

Third party application logging on remote server is an IOS XR logging feature that efficiently manages large volumes of log messages and high rates of logging by forwarding logs generated by third-party applications to a remote server.

- Before Release 24.4.1, the router couldn't efficiently handle large volumes of log messages and high rates of logging because it stored remote syslog logging configuration information in the router's shared memory.
- From Release 24.4.1 onwards, the router processes only native container logs, which can be viewed using the show-logging buffer command. The router no longer processes third party application logs and sends them directly to the remote syslog server.

The XR syslog module processes only native container logs, which can be viewed using the **show-logging buffer** command. Third-party application logs are sent directly to the remote syslog server and are no longer processed by XR. For example, while you can view NOSi application logs using the **show logging** command, XR will no longer process third-party application logs going forward.

Limitations and restrictions for third-party applications logging on remote server

These are the limitations and restrictions for third-party applications logging on remote server:

- You cannot simultaneously enable both "System Log Facility" and "logging container all" configurations.
- You cannot simultaneously enable both "TLS-based remote syslog" and "logging container all" configurations.

Configure third-party applications logging on remote server

You must perform these steps to configure third-party applications logging on remote server:

Procedure

Step 1 Configure the logging to a remote server feature with the **logging source-interface** command to send syslog messages to an external syslog server. For configuration information, see Configuring Logging to a Remote Server.

The kernel level refers to operations or functionalities that are executed within the kernel of an operating system. When you enable kernel-level logging in a syslog configuration, you are primarily instructing the system to log messages that are related to kernel operations.

Step 2 Enable third-party software container logging with the **logging container all** command to capture the logs from all third-party container logs.

Example:

```
Router# configure
Router(config)# logging container all
Router(config)# commit
```

Note
If you don't configure **logging container all** command, the Cisco IOS XR software doesn't forward the logs to the remote syslog server.

Configuring System Logging

Perform the tasks in this section for configuring system logging as required.

Configuring Logging to the Logging Buffer

Syslog messages can be sent to multiple destinations including an internal circular buffer known as logging buffer. You can send syslog messages to the logging buffer using the **logging buffered** command.

Configuration Example

This example shows the configuration for sending syslog messages to the logging buffer. The size of the logging buffer is configured as 3000000 bytes. The default value for the size of the logging buffer is 2097152 bytes.

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# logging buffered 3000000
RP/0/RP0/CPU0:Router(config)# commit
```

Configuring Logging to a Remote Server

Syslog messages can be sent to destinations other than the console, such as logging buffer, syslog servers, snmp servers and terminal lines. You can send syslog messages to an external syslog server by specifying the ip address or hostname of the syslog server using the **logging** command. Also you can configure the syslog facility in which syslog messages are send by using the **logging facility** command.

The following table list the features supported by Cisco IOS XR Software to help managing syslog messages sent to syslog servers.

Table 4: Features for Managing Syslog Messages

Features	Description
UNIX system log facility	Facility is the identifier used by UNIX to describe the application or process that submitted the log message. You can configure the syslog facility in which syslog messages are sent by using the **logging facility** command.

Features	Description
Hostname prefix logging	Cisco IOS XR Software supports hostname prefix logging. When enabled, hostname prefix logging appends a hostname prefix to syslog messages being sent from the router to syslog servers. You can use hostname prefixes to sort the messages being sent to a given syslog server from different networking devices. Use the **logging hostname** command to append a hostname prefix to syslog messages sent to syslog servers
Syslog source address logging	By default, a syslog message sent to a syslog server contains the IP address of the interface it uses to leave the router. Use the **logging source-interface** command to set all syslog messages to contain the same IP address, regardless of which interface the syslog message uses to exit the router.

Configuration Example for Logging to Syslog Server

This example shows the configuration for sending syslog messages to an external syslog server. The ip address 209.165.201.1 is configured as the syslog server.

```
Router# configure
Router(config)# logging 209.165.201.1 vrf default
Router(config)# logging facility kern (optional)
Router(config)# logging hostnameprefix 203.0.113.1 (optional)
Router(config)# logging source-interface HundredGigE 0/0/1/0 (optional)
Router(config)# commit
```

Amongst other options, **all** and **none** are provided under the **logging severity** command form. If you enable **all** or **none**, all or no severity logs are logged to the syslog server, respectively. This configuration persists even when you enable a specific operator type.

Configuration Example for Logging to SNMP Server

This example shows the configuration for sending syslog messages to an SNMP server. The logging trap command is used to limit the logging of messages sent to the snmp servers based on severity.

```
Router# configure
Router(config)# snmp-server traps syslog
Router(config)# logging trap warnings
Router(config)# commit
```

For more information on SNMP server configurations, see the *Configuring Simple Network Management Protocol* chapter in the *System Management Configuration Guide for Cisco NCS 560 Series Routers*

Related Topics

- Configuring Logging to the Logging Buffer, on page 4
- Configuring Logging to Terminal Lines, on page 6

Configuring Logging to Terminal Lines

By default syslog messages will be sent to the console terminal. But, syslog messages can also be send to terminal lines other than the console. You can send syslog messages to the logging buffer using the **logging monitor** command.

Configuration Example

This example shows the configuration for sending syslog messages to terminal lines other than console. In this example, severity level is configured as critical. The terminal monitor command is configured to display syslog messages during a terminal session. The default severity level is debugging.

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# logging monitor critical
RP/0/RP0/CPU0:Router(config)# commit
RP/0/RP0/CPU0:Router# terminal monitor
```

Modifying Logging to Console Terminal

By default syslog messages will be sent to the console terminal. You can modify the logging of syslog messages to the console terminal

Configuration Example

This example shows how to modify the logging of syslog messages to the console terminal.

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# logging console alerts
RP/0/RP0/CPU0:Router(config)# commit
```

Modifying Time Stamp Format

By default, time stamps are enabled for syslog messages. Time stamp is generated in the month day HH:MM:SS format indicating when the message was generated.

Configuration Example

This example shows how to modify the time-stamp for syslog and debugging messages.

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# service timestamps log datetime localtime msec or service
timestamps log uptime
RP/0/RP0/CPU0:Router(config)# service timestamps debug datetime msec show-timezone or service
 timestamps debug uptime
RP/0/RP0/CPU0:Router(config)# commit
```

Suppressing Duplicate Syslog Messages

Suppressing duplicate messages, especially in a large network, can reduce message clutter and simplify the task of interpreting the log. The duplicate message suppression feature substantially reduces the number of duplicate event messages in both the logging history and the syslog file.

Configuration Example

This example shows how to suppress the consecutive logging of duplicate syslog messages.

```
RP/0/RP0/CPU0:Router# configure
RP/0/RP0/CPU0:Router(config)# logging suppress duplicates
RP/0/RP0/CPU0:Router(config)# commit
```

Archiving System Logging Messages to a Local Storage Device

Syslog messages can also be saved to an archive on a local storage device, such as the hard disk or a flash disk. Messages can be saved based on severity level, and you can specify attributes such as the size of the archive, how often messages are added (daily or weekly), and how many total weeks of messages the archive will hold. You can create a logging archive and specify how the logging messages will be collected and stored by using the **logging archive** command.

The following table lists the commands used to specify the archive attributes once you are in the logging archive submode.

Table 5: Commands Used to Set Syslog Archive Attributes

Features	Description
archive-length weeks	Specifies the maximum number of weeks that the archive logs are maintained in the archive. Any logs older than this number are automatically removed from the archive.
archive-size size	Specifies the maximum total size of the syslog archives on a storage device. If the size is exceeded then the oldest file in the archive is deleted to make space for new logs.
device {disk0 | disk1 | harddisk}	Specifies the local storage device where syslogs are archived. By default, the logs are created under the directory device**/var/log**. If the device is not configured, then all other logging archive configurations are rejected. We recommend that syslogs be archived to the harddisk because it has more capacity than flash disks.
file-size size	Specifies the maximum file size (in megabytes) that a single log file in the archive can grow to. Once this limit is reached, a new file is automatically created with an increasing serial number.
frequency {daily | weekly}	Specifies if logs are collected on a daily or weekly basis.
severity severity	Specifies the minimum severity of log messages to archive. All syslog messages greater than or equal to this configured level are archived while those lesser than this are filtered out.

Configuration Example

This example shows how to save syslog messages to an archive on a local storage device.

```
Router#conf t
Router(config)#logging archive
Router(config-logging-arch)#device disk1
Router(config-logging-arch)#frequency weekly
Router(config-logging-arch)#severity warnings
Router(config-logging-arch)#archive-length 6
Router(config-logging-arch)#archive-size 50
Router(config-logging-arch)#file-size 10
Router(config-logging-arch)#commit
```

Platform Automated Monitoring

Platform Automated Monitoring (PAM) is a system monitoring tool integrated into Cisco IOS XR software image to monitor the following issues:

- process crashes
- memory leaks
- CPU hogs
- tracebacks
- disk usage

PAM is enabled by default. When the PAM tool detects any of these system issues, it collects the required data to troubleshoot the issue, and generates a syslog message stating the issue. The auto-collected troubleshooting information is then stored as a separate file in `harddisk:/cisco_support/` or in `/misc/disk1/cisco_support/ directory`.

PAM Events

When PAM detects a process crash, traceback, potential memory leak, CPU hog, a full file system, , it automatically collects logs and saves these logs (along with the core file in applicable cases) as a *.tgz* file in harddisk:/cisco_support/ or in /misc/disk1/cisco_support/ directory. PAM also generates a syslog message with severity level as warning, mentioning the respective issue.

The format of the .tgz file is: *PAM-<platform>-<PAM event>-<node-name>-<PAM process>-<YYYYMMDD>-<checksum>.tgz*.For example, *PAM--crash-xr_0_RP0_CPU0-ipv4_rib-2016Aug16-210405.tgz* is the file collected when PAM detects a process crash.

Because PAM assumes that core files are saved to the default archive folder (harddisk:/ or /misc/disk1/), you must not modify the location of core archive (by configuring exception filepath) or remove the core files generated after PAM detects an event. Else, PAM does not detect the process crash. Also, once reported, the PAM does not report the same issue for the same process in the same node again.

For the list of commands used while collecting logs, refer Files Collected by PAM Tool, on page 11.

The sections below describe the main PAM events:

Crash Monitoring

The PAM monitors process crash for all nodes, in real time. This is a sample syslog generated when the PAM detects a process crash:

```
RP/0/RP0/CPU0:Aug 16 21:04:06.442 : logger[69324]: %OS-SYSLOG-4-LOG_WARNING : PAM detected
```

```
 crash for ipv4_rib on 0_RP0_CPU0.
All necessary files for debug have been collected and saved at
0/RP0/CPU0 : harddisk:/cisco_support/PAM--crash-xr_0_RP0_CPU0-ipv4_rib-2016Aug16-210405.tgz

Please copy tgz file out of the router and send to Cisco support. This tgz file will be
removed after 14 days.)
```

Traceback Monitoring

The PAM monitors tracebacks for all nodes, in real time. This is a sample syslog generated when the PAM detects a traceback:

```
RP/0/RP0/CPU0:Aug 16 21:42:42.320 : logger[66139]: %OS-SYSLOG-4-LOG_WARNING : PAM detected
 traceback for ipv4_rib on 0_RP0_CPU0.
All necessary files for debug have been collected and saved at
0/RP0/CPU0 :
harddisk:/cisco_support/PAM--traceback-xr_0_RP0_CPU0-ipv4_rib-2016Aug16-214242.tgz
Please copy tgz file out of the router and send to Cisco support. This tgz file will be
removed after 14 days.)
```

Memory Usage Monitoring

The PAM monitors the process memory usage for all nodes. The PAM detects potential memory leaks by monitoring the memory usage trend and by applying a proprietary algorithm to the collected data. By default, it collects top output on all nodes periodically at an interval of 30 minutes.

This is a sample syslog generated when the PAM detects a potential memory leak:

```
RP/0/RP0/CPU0:Aug 17 05:13:32.684 : logger[67772]: %OS-SYSLOG-4-LOG_WARNING : PAM detected
 significant memory increase
(from 13.00MB at 2016/Aug/16/20:42:41 to 28.00MB at 2016/Aug/17/04:12:55) for
pam_memory_leaker on 0_RP0_CPU0.
All necessary files for debug have been collected and saved at
0/RP0/CPU0 :
harddisk:/cisco_support/PAM--memory_leak-xr_0_RP0_CPU0-pam_memory_leaker-2016Aug17-051332.tgz

(Please copy tgz file out of the router and send to Cisco support. This tgz file will be
removed after 14 days.)
```

CPU Monitoring

The PAM monitors CPU usage on all nodes periodically at an interval of 30 minutes. The PAM reports a CPU hog in either of these scenarios:

- When a process constantly consumes high CPU (that is, more than the threshold of 90 percentage)
- When high CPU usage lasts for more than 60 minutes

This is a sample syslog generated when the PAM detects a CPU hog:

```
RP/0/RP0/CPU0:Aug 16 00:56:00.819 : logger[68245]: %OS-SYSLOG-4-LOG_WARNING : PAM detected
 CPU hog for cpu_hogger on 0_RP0_CPU0.
All necessary files for debug have been collected and saved at 0/RP0/CPU0 :
harddisk:/cisco_support/PAM--cpu_hog-xr_0_RP0_CPU0-cpu_hogger-2016Aug16-005600.tgz
(Please copy tgz file out of the router and send to Cisco support. This tgz file will be
removed after 14 days.)
```

```
RP/0/RP0/CPU0:Jun 21 15:33:54.517 : logger[69042]: %OS-SYSLOG-1-LOG_ALERT : PAM detected
ifmgr is hogging CPU on 0_RP0_CPU0!
```

File System Monitoring

The PAM monitors disk usage on all nodes periodically at an interval of 30 minutes. This is a sample syslog generated when the PAM detects that a file system is full:

```
RP/0/RP0/CPU0:Jun 20 13:59:04.986 : logger[66125]: %OS-SYSLOG-4-LOG_WARNING : PAM detected
 /misc/config is full on 0_1_CPU0
(please clean up to avoid any fault caused by this). All necessary files for debug have
been collected and saved at
0/RP0/CPU0 : harddisk:/cisco_support/PAM--disk_usage-xr_0_1_CPU0-2016Jun20-135904.tgz
(Please copy tgz file out of the router and send to Cisco support. This tgz file will be
removed after 14 days.)
```

Disable and Re-enable PAM

The PAM tool consists of three monitoring processes—monitor_cpu.pl, monitor_crash.pl, and monitor_show_logging.pl.

Before disabling or re-enabling the PAM, use these options to check if the PAM is installed in the router:

- From Cisco IOS XR Command Line Interface:

```
Router# show pam status
Tue Jun 14 17:58:42.791 UTC
PAM is enabled
```

- From router shell prompt:

```
Router# run ps auxw|egrep perl

root     12559  0.0  0.0  57836 17992 ?        S    Apr24   0:00 /usr/bin/perl
/pkg/opt/cisco/pam//pam_plugin.pl
```

Disable PAM

To disable PAM agent systemwide, execute the following command from XR EXEC mode:

```
Router# disable-pam
```

Re-enable PAM

To re-enable PAM agent systemwide, execute the following command from XR EXEC mode:

```
Router# enable-pam
```

Data Archiving in PAM

At any given point of time, PAM does not occupy more than 200 MB of harddisk: space. If more than 200 MB is needed, then PAM archives old files and rotates the logs automatically.

The PAM collects CPU or memory usage (using **top -b -n1** command) periodically at an interval of 30 minutes. The files are saved under harddisk:/cisco_support/ directory with the filename as <node name>.log (for example, harddisk:/cisco_support/xr-0_RP0_CPU0.log). When the file size exceeds the limit of 15MB, the

file is archived (compressed) into .tgz file, and then rotated for a maximum of two counts (that is, it retains only two .tgz files). The maximum rotation count of .tgz files is three. Also, the old file (ASCII data) is archived and rotated if a node is reloaded. For example, xr-0_RP0_CPU0.log is archived if RP0 is reloaded.

You must not manually delete the core file generated by the PAM. The core file is named as *<process name>_pid.by_user.<yyyymmdd>-<hhmmss>.<node>.<checksum>.core.gz*.

Files Collected by PAM Tool

The table below lists the various PAM events and the respective commands and files collected by the PAM for each event.

You can attach the respective.*tgz* file when you raise a service request (SR) with Cisco Technical Support.

Note Starting from Cisco IOS XR Release 25.1.1, the core file format changes from. `.gz` to `.lz4` and the default core file location is changed from `/misc/disk1` to `/misc/disk1/coredumps` .

Event Name	Commands and Files Collected by PAM
Process crash	• **show install active** • **show platform** • **show version** • core (gz) file • core.txt file
Process traceback	• **show dll** • **show install active** • **show logging** • **show platform** • **show version**
Memory leak	• **show install active** • **show platform** • **show version** • core (gz) file • dumpcore running • continuous memory usage snapshots

Event Name	Commands and Files Collected by PAM
Show logging event	• **show install active** • **show logging** • **show platform** • **show version** • core (gz) file • core.txt file
CPU hog	• **follow process** • **pstack** • **show dll** • **show install active** • **show platform** • **show version** • **top -H** • core (gz) file • CPU usage snapshots
Disk usage	• **show install active** • **show platform** • **show version** • console log • core (gz) file • Disk usage snapshots

Insecure features warning syslog messages

Insecure features warning syslog messages are system notifications that display warning syslog message when you run an insecure command.

Table 6: Feature History Table

Feature Name	Release Information	Feature Description
Insecure features warning syslog messages	Release 25.4.1	Cisco IOS XR software warns you with a syslog message each time you use an insecure command and repeats the warning every 30 days until you remove the command. This helps you identify potential security risks and suggests safer alternatives to improve your network security. Cisco will systematically deprecate and eventually remove these insecure features and protocols in future IOS XR releases. For more information on insecure commands and their alternatives, see Feature deprecation phasing out insecure capabilities.

When you configure or execute an insecure command, the Cisco IOS XR software displays the following types of messages:

- **Periodic warning**: When you configure an insecure command, the Cisco IOS XR software displays a syslog message when you first execute an insecure command, and it repeats the warning syslog message every 30 days until you remove the insecure command.
- **One-Time warning**: When you execute an insecure command, the Cisco IOS XR software displays a syslog message after you execute an insecure command.
- **Informational message**: When you unconfigure an insecure command, the Cisco IOS XR software displays a syslog message after you remove an insecure command.

You cannot disable warning syslog messages. However, you can filter or change the syslog severity level to control which messages you can see.

You can see the messages on both active and standby route processors in a dual-RP system.

From Release 25.4.1 the insecure commands are deprecated and will be removed in the future release.

These documents list all features planned for removal, including insecure commands, and provide recommended secure alternatives to help you maintain network security and compliance.

- Feature deprecation phasing out insecure capabilities
- Feature deprecation and removal details
- Feature removal and suggested alternatives

Warning syslog message format

The Cisco IOS XR software display a warning syslog message in the following format:

- **For configuring an insecure configuration or performing an insecure action**:

```
%INFRA-WARN_INSECURE-4-INSECURE_FEATURE_WARN : Feature <insecure_feature_name> utilized
 or configured. This feature is deprecated as it is known to be insecure; it will be
removed in a future release. <additional_details>
```

- **For unconfigure an insecure feature**:

```
%INFRA-WARN_INSECURE-6-INSECURE_CONFIG_REMOVED : Insecure feature <insecure feature
name> configuration removed.
```

Table 7: Insecure feature or protocol

Insecure feature or protocol	Description
Insecure feature name	Displays the insecure feature short name. For more information on insecure commands and their alternatives, see Feature deprecation phasing out insecure capabilities.
Additional details	Displays additional details such as a secure alternative to use instead of the insecure command or configuration. For more information on insecure commands and their alternatives, see Feature deprecation phasing out insecure capabilities.

Some features may not have an alternate suggestion. In these cases, the one-time warning syslog message will not include additional details.

Persistent logging

Persistent logging is a logging capability that

- writes incoming syslog messages to both the in-memory logging buffer and a persistent file on local storage
- restores log messages from the persistent file after a node reload, router load, or RP failover, and
- ensures that syslog messages generated before such events remain available for troubleshooting.

Table 8: Feature History Table

Feature Name	Release Information	Feature Description
Persistent logging	Release 26.3.1	This feature preserves the contents of the IOS XR logging buffer across node reloads, router reloads, and RP failovers. Syslog messages are written to both the in-memory logging buffer and a persistent file on local storage. After a reload or failover, the system restores the logging buffer from persistent storage so that retained log messages before the event and current logs, are displayed. This feature introduces these changes: CLI: • A new keyword, **backup-period**, has been added to the **logging buffered** command. • A new command, **show syslog persistent status**, has been introduced.

Persistent logging

Prior to Cisco IOS XR Release 26.3.1, the log messages were stored only in a shared-memory circular buffer on the Active RP. That buffer survived process restarts, but it did not survive node or router reloads. As a result, log messages generated before a reload or failover were not available for troubleshooting after the system came back up.

The persistent logging feature, introduced in Cisco IOS XR Release 26.3.1, preserves the contents of the system logging buffer across node reloads, router reloads, and RP failovers. This feature is enabled by default.

The existing **logging buffered** configuration determines the persistent file size, while the new **logging buffered backup-period** command controls how often logs are flushed to persistent storage.

To view the status of persistent logging buffers, use the **show syslog persistent status** command.

When the configured size limit is reached, the oldest log messages are overwritten.

Benefits of persistent logging

Persistent logging provides these benefits:

- Preserves log history across reload and failover events,
- Improves post-event troubleshooting.

- Keeps the **show logging** workflow unchanged.
- Maintains log continuity on systems that use active and standby RPs.

How persistent logging works

Persistent logging preserves syslog messages by writing them to both volatile memory and persistent storage, ensuring logs are retained during reloads and failover events. This process maintains log continuity for troubleshooting.

The key components involved in the process are:

- Active route processor (RP): Receives syslog messages and maintains a logging buffer in shared memory.
- Persistent storage: Stores log messages in a file that survives reloads and failover events.
- Standby RP: On systems with high availability, receives mirrored log updates and maintains its own buffer and file.

Workflow

These stages describe how persistent logging works:

1. The active RP receives syslog messages and writes them to the shared-memory logging buffer.
2. The system copies the same log messages to a persistent file on local storage.
3. The system batches log message writes to persistent storage to reduce disk write frequency. The system flushes the batch when:
   - the backup period timer expires
   - the batching buffer reaches its maximum size
   - the logging buffer size changes
   - the backup period changes
   - a graceful reload event occurs
4. On systems with active and standby RPs, the active RP mirrors log updates to the standby RP. The standby RP maintains its own shared-memory buffer and persistent file.
5. After a reload or RP failover, the system repopulates the shared-memory logging buffer from the persistent file. The **show logging** command then displays retained pre-event logs and current logs.

Result

Graceful reload and failover events flush pending log data to disk before the event completes, ensuring logs are preserved.

Ungraceful events are also supported, but the newest log messages can be lost if they have not yet been flushed to persistent storage.

Persistent logging defaults and limits

This table lists the default values and storage limits for persistent logging.

Item	Value
Persistent logging state	Enabled by default
Default backup period	10 seconds
Backup period range	10 msec to 60 seconds
Default logging buffer size	2 MB
Approximate default message count	2545 messages
Maximum configured size	125000000 bytes
Approximate maximum message count	~150k messages
Default persistent device location	`/var/log`
Persistent device location	Platform-dependent
Persistent file location and filename	Platform-defined and not user-configurable
File behavior when full	Circular overwrite of oldest messages
Persistent file size control	Uses the existing **logging buffer** <*size/entries-count*> configuration

Persistent logging on active and standby RPs

On systems that use active and standby RPs, persistent logging works as follows:

- The active RP writes syslog messages to its local shared-memory logging buffer and persistent file.
- The active RP mirrors log updates to the standby RP to maintain synchronization.
- The standby RP maintains its own shared-memory logging buffer and persistent file, ensuring redundancy.
- When the standby RP boots, the system copies the persistent file from the active RP and then continues to update the standby RP with subsequent syslog messages.
- After failover, the newly active RP as an up-to-date log history, providing seamless log continuity and reliable historical data.

Guidelines for persistent logging

Remember these guidelines for persistent logging:

- Use a shorter backup period to reduce the amount of log data that can be lost during an ungraceful reload.
- Use a longer backup period to reduce disk write frequency and help protect the storage device from excessive wear.
- Plan for a small loss window during an ungraceful reload because batched log messages might not yet be written to disk.
- Use the existing **logging buffer** configuration to control the persistent file size.
- Expect the system to overwrite the oldest log messages when the persistent file reaches its size limit.
- Expect the persistent device location and filename to remain platform-defined.

Restrictions for persistent logging

To ensure reliable and predictable persistent logging behavior, follow these restrictions:

- Persistent logging depends on local storage. The feature writes logs to disk, so the backup period setting balances log retention.
- The persistent file size depends on the configured logging buffer size.
- Large bursts of syslog messages can trigger an early flush when the batching buffer reaches its size limit.
- Graceful reload events flush the batching buffer before reload completes.
- Ungraceful reload events can lose the newest log messages that remain in the batching buffer.

Configure persistent logging

Use this task to control the flush interval for persistent logging, balancing log retention and disk write frequency.

Adjusting the persistent logging interval can help tailor log data retention during unexpected reloads and manage how frequently logs are written to disk.

Before you begin

Decide whether your deployment needs a shorter log retention window for unexpected reloads or a lower disk write frequency.

Follow these steps to configure the persistent logging backup period:

Procedure

Step 1 Enter the **logging buffer backup-period** *value* <*sec*|*msec*> command to set the backup period.

Example:

```
Router(config)# logging buffer backup-period 30
```

The **backup-period** value ranges from 10 millseconds to 60 seconds.

Use a shorter value if you want to reduce the amount of recent log data that can be lost during an ungraceful reload.

Use a longer value if you want to reduce the frequency of writes to persistent storage.

The system flushes the current batch of syslog messages to disk and restarts the backup timer with the new value.

Step 2 Execute the **show logging** command after a reload or RP failover to verify that the log messages before the reload or failover are retained in the logging buffer contents.

Example:

```
Router# show logging
```

Confirm that the output shows the persistent device location and persistent file name, if supported on your platform.

The **show logging** output includes retained log messages that the system restored from persistent storage.

Step 3 Run the **show syslog persistent status** command to verify the persistent file details.

Example:

```
Router# show syslog persistent status
Thu Aug 6 11:14:32.464 UTC
node: node0_RP0_CPU0
Persistent file details:
Filename: /var/log/xr_syslog/persistent/messages.log
Maximum Size: 2731 messages (2097432 bytes)
Message Write Index: 260 messages (199704 bytes)
Dropped Messages: 0
Invalid Messages: 14
No of Wraps: 0
Backup Period: 10000 milliseconds
```

The system retains log messages across reloads and failovers according to your configured backup period.

What to do next

Use the **clear logging** command to remove retained log data from the active and standby RPs.