ERICSSON

Logging

SYSTEM ADMINISTRATOR GUIDE

82/1543-CRA 119 1170/1-V1 Uen H

Contents

1 Logging

This document provides an overview of logging features, describes the tasks used to configure them, and provides configuration examples and detailed descriptions of the commands used to configure logging features using the SmartEdgerouter .

Note: In the following descriptions, the term controller card applies to the Cross-Connect Route Processor (XCRP4) Controller card, including the controller carrier card unless otherwise noted.

The term controller carrier card refers to the controller functions on the carrier card within the SmartEdge 100 chassis The term I/O carrier card refers to the traffic card functions on the carrier card; these functions are compatible with the similar functions that are implemented on the traffic card that are supported on all other SmartEdge routers.

This document applies to both the Ericsson SmartEdge® and SM family routers. However, the software that applies to the SM family of systems is a subset of the SmartEdge OS; some of the functionality described in this document may not apply to SM family routers.

For information specific to the SM family chassis, including line cards, refer to the SM family chassis documentation.

For specific information about the differences between the SmartEdge and SM family routers, refer to the Technical Product Description *SM Family of Systems* (part number 5/221 02-CRA 119 1170/1) in the **`Product Overview`** folder of this Customer Product Information library.

1.1 About Logging

The operating system contains two log buffers: main and debug. By default, messages are stored in the main log. If the system restarts, for example as a result of a logging daemon or system error, and the logger daemon shuts down and restarts cleanly, the log buffers are saved to the **/md/loggd_dlog.bin** for the main log buffer, and the **/md/loggd_ddbg.bin** for the debug log buffer. You can view the contents of the main log files that are saved using the **`show log`** command (in any mode).

Note: The debug buffer is not fully supported in this release. You cannot use the **`show log`** command (in any mode) to display the contents of the debug buffer. To view all log messages, enable the **`logging debug`** command (in global configuration mode), so that the contents of the debug buffer can be displayed using the **`show log`** command (in exec mode). Be aware that enabling the **`logging debug`** command can quickly fill up the log buffer with debug and non debug messages. To prevent the main buffer from filling up with debug messages and overwriting other more significant messages, disable the **`logging debug`** command, (in context configuration mode).

By default, log messages for local contexts are displayed in real time on the console; non-local contexts are not displayed in real time on the console. To change this behavior, and display messages in real time, use the **`logging console`** command (in context configuration mode). However, log messages can be displayed in real time from any Telnet session using the **`terminal monitor`** command (in exec mode). For more information on the **`terminal monitor`** command, see *Command List*.

In large installations, it is convenient to have all systems log to a remote machine for centralized management and to save space on the device. The operating system uses the UNIX syslog facility for this purpose, and can send log messages to multiple machines concurrently. Logging can be constrained to events occurring on a specific circuit.

All log messages contain a numeric value indicating the severity of the event or condition that caused the message to be logged. Many log messages are normal and do not indicate a system problem.

Table 1 lists event severity levels in log messages and their respective descriptions.

Table 1 Event Severity Levels in Log Messages

Value	Severity Level	Description
0	emergencies	Panic condition—the system is unusable.
1	alerts	Immediate administrator intervention is required.
2	critical	Critical conditions have been detected.
3	errors	An error condition has occurred.
4	warnings	A potential problem exists.
5	notifications	Normal, but significant, events or conditions exist.
6	informational	Informational messages only; no problem exists.
7	debugging	Output from an enabled system debugging function.

1.2 How to Configure Logging

This section describes how to configure logging in several different scenarios.

1.2.1 Configure Optional Global Logging Features

To configure optional global logging features, perform the tasks described in Table 2; enter all commands in global configuration mode.

Table 2 Configure Optional Global Logging Features

Task	Root Command	Notes
Enable the display of logged system event messages with a millisecond resolution timestamp.	*logging timestamp milli second*	
Enables the logger to send logging and debug messages from the active controller card to the standby controller card.	*logging active*	Enter the **no** form of this command to disable this feature.
Enables the filtering of debug messages for valid circuits only.	*logging cct-valid*	Enter this command in global configuration mode.
Enables the logger to send logging and debug messages from the standby controller card to the active controller card.	*logging standby*	Use the **no** form of this command to disable this feature.
Enables the display of all debug messages in the main log buffer.	*logging debug*	Use the **no** form of this command to prevent the debug messages from being sent to the main log buffer.

1.2.2 Configure Optional Context-Specific Logging Features

To configure optional context-specific logging features, perform the tasks described in Table 3; enter all commands in context configuration mode, unless otherwise noted.

Table 3 Configure Optional Context-Specific Logging Features

Task	Root Command	Notes
Isolate events from certain facilities in the logs and trim the flow of information.	*logging filter*	
Enable event logging messages to the console.	*logging console*	

Table 3 Configure Optional Context-Specific Logging Features

Task	Root Command	Notes
Enable event logging messages to a file.	*logging file*	You can configure up to four log files per context.
Enable the logging of system events to a remote syslog server that is reachable within the current context.	*logging syslog*	You can configure up to four syslog servers per context.

The following example configures the system to remotely log all system messages to a network syslog server. Information to forward packets to the **10.1.1.1** address specified for the syslog host is derived from routing tables specific to the **NewContext** context:

```
[local]Redback(config)#context NewContext
[local]Redback(config-ctx)#logging syslog 10.1.1.1
```

The following example shows a configuration where log messages are sent to a syslog server (**198.168.148.99**) in the **local** context using the syslog facility, **local6**, and to another syslog server (**198.168.145.99**) in the **green** context using the syslog facility, **local3:**

```
[local]Redback(config)#context local
[local]Redback(config-ctx)#logging sys 198.168.148.99 facility local6
[local]Redback(config-ctx)#exit
[local]Redback(config)#context green
[local]Redback(config-ctx)#logging sys 198.168.145.99 facility local3
```

1.3 Logging Operations Tasks

Table 4 lists the logging operations tasks. Enter the **show** commands in any mode; enter all other commands in exec mode.

Table 4 Logging Operations Tasks

Task	Root Command
Clear the system event log buffer.	*clear log*
Clear the contents of the nonvolatile memory (NVRAM) on the active controller card to which you are connected.	*clear system nvlog*
Enable the generation of debugging messages for the logging facility (logger).	*debug logger*
Enable the generation of debugging messages for the logging facility (logger) RCM.	*debug logger-rcm*
Saves one of the internal event log buffers to the flash file system.	*save log*

Table 4 Logging Operations Tasks

Task	**Root Command**
Display information about system event logs or a previously saved log file.	*show logging*
Display statistics about the system logger, including logger uptime, number of logged messages, number of logged filter messages, and number of logged rate-limited messages.	*show logging*
Display the contents of the NVRAM on the active controller card to which you are connected.	*show system nvlog*

2 ISP Logging

The in-service performance (ISP) log is a file stored in the flash memory of the SmartEdge router . It collects information about predefined system events that can have a potential impact on applications and enable support representatives to perform root-cause analysis and troubleshooting on the SmartEdge router . You can view the ISP log in the CLI by using the **show isp-log** command, or you can extract the ISP log from /flash/.isp.log by using the **copy** command. The ISP log is persistent across switchovers and reboots.

When the ISP log file reaches the size limit you set with the **isp-log size** command, the system stops writing ISP log entries in the file, logs an entry in the ISP file stating that the file is full, and displays the following system error messages:

- `%SYSLOG-3-ERROR: ISP logging disabled due to file size limit being reached`
- `%ISP-3-ERR: ISP Log is Full, current /flash/.isp.log file size` *`file size`* `exceeds max file size` *`max file size`*
  where *`file size`* is the file size of the log file and *`max file size`* is the maximum file size set by the **isp-log size** command.

To resume logging entries in the ISP log file, you must either:

- Extract the ISP log file using the updated **copy** command with the new **clear** keyword. When the ***src-url*** argument is the location of the ISP log (/flash/.isp.log), the **clear** keyword clears the contents of the local ISP log file after the file is copied successfully. If the system stopped logging ISP entries because the ISP file had reached the size limit, this keyword causes the system to resume ISP logging.
- Clear the ISP log file using the **clear isp-log** command.

If you disable the ISP log or change the size limit, the system removes any existing ISP log file. To save an existing ISP log file before disabling ISP logging, use the **copy** command to extract the file from the system.

You can use the information in the ISP log to manually compute system downtime and other statistics or, in the event of an issue, you can send the extracted file to your support representative for analysis.

The ISP log tracks and displays the following information:

- Event type. See Table 7 for more on the specific event types in the ISP log file.
- Application name. Events that are not application specific use the application name, "system."

- Event time stamp. The time the event occurred, in UDC format.
- Event information. Additional details of the source of the event. See Table 7 for the details that display in the ISP log for certain events:
- Trigger Method - If a user performed the action, the ISP log records the trigger method as "manual." If the system performed the action, the ISP log records the trigger method as "auto."
- System Uptime. Time since the system last rebooted, in seconds.
- Comment - Displayed if a user added a comment using the **`isp-log add`** ***`comment`*** CLI command.

2.1 Configure ISP Log Size

To configure the ISP log size, enter the *isp-log size* command in global configuration mode.

When the ISP log file reaches the size limit you set, the system stops writing ISP log entries in the file, logs an entry in the ISP file stating that the file is full, and displays a system error message.

2.2 Perform Exec-Level Commands on the ISP Log

To perform exec-level commands on the ISP log, perform the tasks described in Table 5; enter all commands in exec mode.

Table 5 Configure ISP Logging Features

Task	Root Command	Notes
Enable ISP logging.	*isp-log*	Enter the **`no`** form of this command to disable this feature.
Add a comment to the ISP log file.	*isp-log add*	
Clear the ISP log file	*clear isp-log*	
Extract the ISP log file	*copy*	Use the **`clear`** keyword with this command to clear the ISP log file after extraction.

2.3 Display the ISP Log and Log File Information

Table 6 lists the commands to display the ISP log and log file information. Enter the **`show`** commands in any mode.

Table 6 Perform ISP Logging Operations

Task	Root Command
Display the ISP log file	*show isp-log*
Display ISP log state information	*show isp-log state*

2.3.1 Event Types

When you view the ISP log using the **`show isp-log`** command or by viewing a file extracted using the **`copy`** command, the log displays a number of event types. Table 7 identifies the event type terms, their descriptions, and additional event information displayed in the ISP log file.

Table 7 Event Types and Information

Event Type	Description	Event Information
node_down	The router went down.	N/A
node_up	The router came up. During a system reboot, process-up and card-up events are logged on each of the processes and line cards that come up. A single node_up event is logged.	N/A
proc_down	A process went down	Process name and instance ID
proc_up	A process came up.	Process name and instance ID
linecard_down	A line card went down.	Slot number and card type
linecard_up	A line card came up.	Slot number and card type
switchover	A switchover occurred.	Reason for switchover (for example, "User requested manual switch")
upgrade	A regular or patch upgrade was performed.	Type of upgrade
heartbeat	Application heartbeat event.	N/A
log_full	The ISP log file reached the maximum file size.	Displays the text, "Max file size reached"
cli_comment	A comment made using the CLI command **`isp-log add`** ***`comment`*** command.	N/A
hostname	The new hostname string.	The new hostname string. Each system should be configured with a unique hostname in order to correlate events to a specific host.

2.3.2 Example: Displaying an ISP Log file

The following is an example of the ISP log file, displayed in the CLI:

```
[local]Redback>show isp-log
16Dec21:10:002009user1;Upgrade;System;2010-01-25 19:32:22 UTC;Regular,
 6.3.1.1;Manual;3419857;
16Dec21:10:002009user1;Node_down;system;2010-01-25 19:32:24 UTC;;Manual;
3419858;
6.3.1.1;Node_up;system;2010-01-25 19:34:36 UTC;;Manual;119;
6.3.1.1;Hostname;System;2010-01-25 19:34:54 UTC;System1;Manual;138;
6.3.1.1;Hostname;System;2010-01-25 19:34:57 UTC;System2;Manual;141;
6.3.1.1;Proc_down;System;2010-01-25 19:36:38 UTC;System3;Manual;243;
6.3.1.1;Proc_up;System;2010-01-25 19:36:51 UTC;System3;Manual;256;
6.3.1.1;Linecard_down;System;2010-01-25 19:38:31 UTC;Slot 1, atm-oc3e-8-port;
Manual;356;
6.3.1.1;Linecard_up;System;2010-01-25 19:38:47 UTC;Slot 1, atm-oc3e-8-port;
Manual;371;
6.3.1.1;Cli_comment;CLI;2010-01-25 19:39:46 UTC;;Manual;431;This is an
 example comment from CLI;
6.3.1.1;Hostname;System;2010-01-25 19:40:35 UTC;System4;Manual;479;
6.3.1.1;Proc_down;System;2010-01-25 19:40:44 UTC;System3;Manual;488;
6.3.1.1;Hostname;System;2010-01-25 19:40:52 UTC;System2;Manual;496;
6.3.1.1;Proc_up;System;2010-01-25 19:40:56 UTC;System3;Manual;500;
6.3.1.1;Node_down;system;2010-01-25 19:41:20 UTC;;Manual;525;
6.3.1.1;Node_up;system;2010-01-25 19:43:31 UTC;;Manual;118;
6.3.1.1;Hostname;System;2010-01-25 19:43:49 UTC;System1;Manual;137;
6.3.1.1;Hostname;System;2010-01-25 19:43:51 UTC;System2;Manual;140;
[local]Redback#
```

The following is an ISP log entry example:

```
6.3.1.1;Linecard_down;System;2010-01-25 19:38:31 UTC;Slot 1,
 atm-oc3e-8-port;Manual;356;
```

Table 8 describes the information in the ISP log entry example.

Table 8 ISP Log Entry Definitions

Example Entry	ISP Log information
`Linecard_down`	Event type
`System`	Application name
`2010-01-25 19:38:31 UTC`	Event time stamp
`Slot 1, atm-oc3e-8-port`	Event information
`Manual`	Trigger Method
`356`	System Uptime
N/A	Comment