Nortel Ethernet Routing Switch 8600

Fault Management

Release: 7.0
Document Revision: 02.01

www.nortel.com

NN46205-705

Nortel Ethernet Routing Switch 8600
Release: 7.0
Publication: NN46205-705
Document release date: 21 December 2009

Contents

Software license 7

New in this release 11
Features 11
Other changes 11

Introduction 13

Fault management fundamentals 15
Remote monitoring 15
RMON Alarms 16
RMON history 18
RMON events 19
RMON statistics 19
Traps and logs 19
Simple Network Management Protocol 19
Overview of traps and logs 20
System Messaging Platform 21
Log message format 21
Log files 24
Log file transfer 27
Link state change control 28

RMON configuration using Enterprise Device Manager 29
Enabling RMON globally 29
Enabling RMON history 31
Variable definitions 31
Disabling RMON history 33
Creating alarms 34
Variable definitions 35
Viewing RMON alarms 36
Variable definitions 36
Viewing RMON events 39
Variable definitions 39
Viewing the RMON log 40
Variable definitions 40

Deleting alarms 40
Creating RMON events (default) 41
Variable definitions 41
Creating events (nondefault) 42
Deleting events 43
Disabling RMON statistics 43

RMON configuration using the CLI 45
Job aid: Roadmap of CLI commands for configuring RMON 45
Configuring RMON 46
Example of configuring RMON 49
Viewing RMON Settings 49
Job aid: Output for show rmon 50
Configuring the switch to capture RMON statistics 50
Procedure steps 50
Variable definitions 50

RMON configuration using the NNCLI 51
Job aid: Roadmap of RMON commands 51
Configuring RMON 53
Variable definitions 54
Viewing RMON settings 56
Job aid: Output for show rmon 57

Log and trap configuration using Enterprise Device Manager 59
SNMP trap configuration 59
Configuring an SNMP host target address 60
Configuring target table parameters 61
Viewing the trap sender table 62
Configuring an SNMP notify table 63
Configuring SNMP notify filter profile table parameters 64
Configuring SNMP notify filter table parameters 65
Enabling SNMP trap logging 66
Viewing SNMP trap logs 66
Log configuration 67
Navigation 67
Configuring the system log 67
Configuring the system log table and severity level mappings 68
Viewing system logs 69
Viewing Enterprise Device Manager logs 70

Log and trap configuration using the CLI 71
SNMP trap configuration 71
Roadmap of SNMP trap CLI commands 72
Configuring SNMP notifications 74
Configuring an SNMP host target address 76

Configuring SNMP target table parameters 78
Configuring an SNMP notify filter table 80
Configuring SNMP interfaces 81
Enabling SNMP trap logging 82
Configuring a UNIX system log and syslog host 83
Log configuration 85
Roadmap of CLI log commands 86
Configuring logging 87
Viewing logs 88
Configuring the remote host address for log transfer 90
Configuring system logging to a PCMCIA 91
Starting system message logging to a PCMCIA card 93
Configuring system message control 94
Extending system message control 95
Configuring CLI logging 96

Log and trap configuration using the NNCLI 99
SNMP trap configuration 99
Roadmap of SNMP trap NNCLI commands 100
Job aid: SNMP configuration in the NNCLI 101
Configuring SNMP notifications 103
Configuring an SNMP host 103
Configuring SNMP target table parameters 106
Configuring an SNMP notify filter table 106
Configuring SNMP interfaces 106
Enabling SNMP trap logging 108
Configuring a UNIX system log and syslog host 109
Log configuration 111
Roadmap of NNCLI log commands 112
Configuring logging 112
Viewing logs 114
Configuring the remote host address for log transfer 115
Configuring system logging to a PCMCIA 116
Starting system message logging to a PCMCIA card 118
Configuring system message control 119
Extending system message control 119
Configuring NNCLI logging 120

Link state change control using Enterprise Device Manager 123
Controlling link state changes using Enterprise Device Manager 123

Link state change control using CLI 125
Controlling link state changes using CLI 125
Example of controlling link state changes 126

Link state change control using NNCLI **127**
Controlling link state changes using NNCLI 127
Example of controlling link state changes 128

RMON alarm variables **129**
RMON alarm reference 129

Software license

This section contains the Nortel Networks software license.

Nortel Networks Inc. software license agreement

This Software License Agreement ("License Agreement") is between you, the end-user ("Customer") and Nortel Networks Corporation and its subsidiaries and affiliates ("Nortel Networks"). PLEASE READ THE FOLLOWING CAREFULLY. YOU MUST ACCEPT THESE LICENSE TERMS IN ORDER TO DOWNLOAD AND/OR USE THE SOFTWARE. USE OF THE SOFTWARE CONSTITUTES YOUR ACCEPTANCE OF THIS LICENSE AGREEMENT. If you do not accept these terms and conditions, return the Software, unused and in the original shipping container, within 30 days of purchase to obtain a credit for the full purchase price.

"Software" is owned or licensed by Nortel Networks, its parent or one of its subsidiaries or affiliates, and is copyrighted and licensed, not sold. Software consists of machine-readable instructions, its components, data, audio-visual content (such as images, text, recordings or pictures) and related licensed materials including all whole or partial copies. Nortel Networks grants you a license to use the Software only in the country where you acquired the Software. You obtain no rights other than those granted to you under this License Agreement. You are responsible for the selection of the Software and for the installation of, use of, and results obtained from the Software.

1. Licensed Use of Software. Nortel Networks grants Customer a nonexclusive license to use a copy of the Software on only one machine at any one time or to the extent of the activation or authorized usage level, whichever is applicable. To the extent Software is furnished for use with designated hardware or Customer furnished equipment ("CFE"), Customer is granted a nonexclusive license to use Software only on such hardware or CFE, as applicable. Software contains trade secrets and Customer agrees to treat Software as confidential information using the same care and discretion Customer uses with its own similar information that it does not wish to disclose, publish or disseminate. Customer will ensure that anyone who uses the Software does so only in compliance with the terms

of this Agreement. Customer shall not a) use, copy, modify, transfer or distribute the Software except as expressly authorized; b) reverse assemble, reverse compile, reverse engineer or otherwise translate the Software; c) create derivative works or modifications unless expressly authorized; or d) sublicense, rent or lease the Software. Licensors of intellectual property to Nortel Networks are beneficiaries of this provision. Upon termination or breach of the license by Customer or in the event designated hardware or CFE is no longer in use, Customer will promptly return the Software to Nortel Networks or certify its destruction. Nortel Networks may audit by remote polling or other reasonable means to determine Customer's Software activation or usage levels. If suppliers of third party software included in Software require Nortel Networks to include additional or different terms, Customer agrees to abide by such terms provided by Nortel Networks with respect to such third party software.

2. Warranty. Except as may be otherwise expressly agreed to in writing between Nortel Networks and Customer, Software is provided "AS IS" without any warranties (conditions) of any kind. NORTEL NETWORKS DISCLAIMS ALL WARRANTIES (CONDITIONS) FOR THE SOFTWARE, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OF NON-INFRINGEMENT. Nortel Networks is not obligated to provide support of any kind for the Software. Some jurisdictions do not allow exclusion of implied warranties, and, in such event, the above exclusions may not apply.

3. Limitation of Remedies. IN NO EVENT SHALL NORTEL NETWORKS OR ITS AGENTS OR SUPPLIERS BE LIABLE FOR ANY OF THE FOLLOWING: a) DAMAGES BASED ON ANY THIRD PARTY CLAIM; b) LOSS OF, OR DAMAGE TO, CUSTOMER'S RECORDS, FILES OR DATA; OR c) DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES (INCLUDING LOST PROFITS OR SAVINGS), WHETHER IN CONTRACT, TORT OR OTHERWISE (INCLUDING NEGLIGENCE) ARISING OUT OF YOUR USE OF THE SOFTWARE, EVEN IF NORTEL NETWORKS, ITS AGENTS OR SUPPLIERS HAVE BEEN ADVISED OF THEIR POSSIBILITY. The forgoing limitations of remedies also apply to any developer and/or supplier of the Software. Such developer and/or supplier is an intended beneficiary of this Section. Some jurisdictions do not allow these limitations or exclusions and, in such event, they may not apply.

4. General

1. If Customer is the United States Government, the following paragraph shall apply: All Nortel Networks Software available under this License Agreement is commercial computer software and commercial computer

software documentation and, in the event Software is licensed for or on behalf of the United States Government, the respective rights to the software and software documentation are governed by Nortel Networks standard commercial license in accordance with U.S. Federal Regulations at 48 C.F.R. Sections 12.212 (for non-DoD entities) and 48 C.F.R. 227.7202 (for DoD entities).

2. Customer may terminate the license at any time. Nortel Networks may terminate the license if Customer fails to comply with the terms and conditions of this license. In either event, upon termination, Customer must either return the Software to Nortel Networks or certify its destruction.
3. Customer is responsible for payment of any taxes, including personal property taxes, resulting from Customer's use of the Software. Customer agrees to comply with all applicable laws including all applicable export and import laws and regulations.
4. Neither party may bring an action, regardless of form, more than two years after the cause of the action arose.
5. The terms and conditions of this License Agreement form the complete and exclusive agreement between Customer and Nortel Networks.
6. This License Agreement is governed by the laws of the country in which Customer acquires the Software. If the Software is acquired in the United States, then this License Agreement is governed by the laws of the state of New York.

New in this release

The following sections detail what's new in *Nortel Ethernet Routing Switch 8600 Fault Management* (NN46205-705)for Release 7.0.

- "Features" (page 11)
- "Other changes" (page 11)

Features

Enterprise Device Manager Replaced the Device Manager configuration information with the Enterprise Device Manager (EDM). Starting with this release, EDM is replacing Device Manager as the graphical user interface. See "RMON configuration using Enterprise Device Manager" (page 29)

Other changes

There are no other changes in this document.

Introduction

This guide to fault management for the Nortel Ethernet Routing Switch 8600 provides information about Remote Monitoring (RMON), traps and logs, controlling link state changes (port flapping), viewing RMON statistics, and RMON alarm variables.

Navigation

- “Fault management fundamentals” (page 15)
- “RMON configuration using Enterprise Device Manager” (page 29)
- “RMON configuration using the CLI” (page 45)
- “RMON configuration using the NNCLI” (page 51)
- “Log and trap configuration using Enterprise Device Manager” (page 59)
- “Log and trap configuration using the CLI” (page 71)
- “Log and trap configuration using the NNCLI” (page 99)
- “Link state change control using Enterprise Device Manager” (page 123)
- “Link state change control using CLI” (page 125)
- “Link state change control using NNCLI” (page 127)
- “RMON alarm variables” (page 129)

Fault management fundamentals

Fault management includes the tools and features available to monitor and manage faults. This section provides overviews for Remote Monitoring (RMON), traps and logs, and link stage changes (port flapping).

Navigation

- "Remote monitoring" (page 15)
- "Traps and logs" (page 19)
- "Link state change control" (page 28)

Remote monitoring

Remote monitoring (RMON) is a management information base (MIB). An MIB is a group of management objects that you can use to obtain or configure values. Use the Simple Network Management Protocol (SNMP) to manipulate the objects in MIB.

You can use the command line interface (CLI), Nortel Networks command line interface (NNCLI), or Device Manager to globally enable RMON for devices on the switch. After you globally enable RMON, you can enable monitoring for individual devices on a port-by-port basis.

RMON has four major functions:

- configure alarms for user-defined events
- collect Ethernet statistics
- log events
- send traps for events

Within Device Manager, you can configure RMON alarms that relate to specific events or variables when you select variables from a list. When you configure the system to send events associated with alarms to trap or log-and-trap, tripped alarms are trapped or logged.

You can view all RMON information using the Device Manager, the CLI, or the NNCLI. You can use any management application that supports SNMP traps to view RMON trap information.

This section includes the following concepts:

- "RMON Alarms" (page 16)
- "RMON history" (page 18)
- "RMON events" (page 19)
- "RMON statistics" (page 19)

RMON Alarms

You can use RMON alarms to alert you if the value of a variable falls outside a designated range.

You can define RMON alarms on any MIB variable that resolves to an integer value but you cannot use string variables, for example, system description, as alarm variables.

All alarms share the following characteristics:

- a defined upper and lower threshold value
- a corresponding rising and falling event
- an alarm interval or polling period

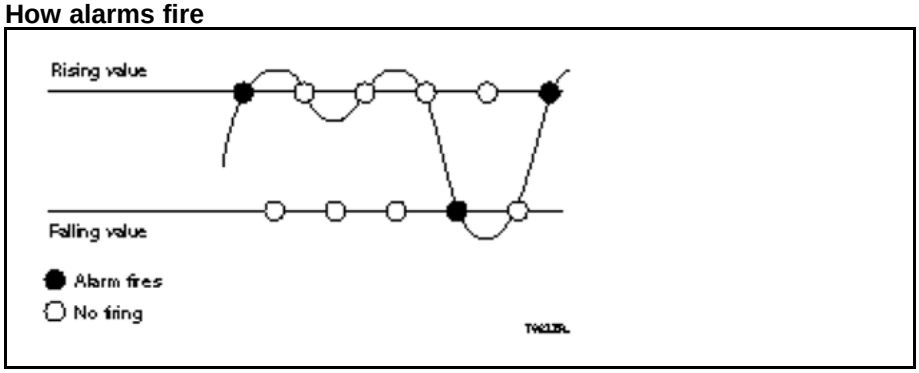
After you activate alarms, you can

- view the activity in a log or a trap log.
- create a script directing the system to sound an audible alert at a console.
- create a script directing the system to send an e-mail.
- create a script directing the system to call a pager

The alarm variable is polled and the result is compared against upper and lower limit values selected when the alarm is created. If either limit is reached or crossed during the polling period, then the alarm fires and generates an event that you can view in the event log or the trap log.

The upper limit of the alarm is the rising value, and the lower limit is the falling value. RMON periodically samples data based upon the alarm interval. During the first interval that the data passes above the rising value, the alarm fires as a rising event. During the first interval that the data drops below the falling value, the alarm fires as a falling event.

Figure 1
How alarms fire



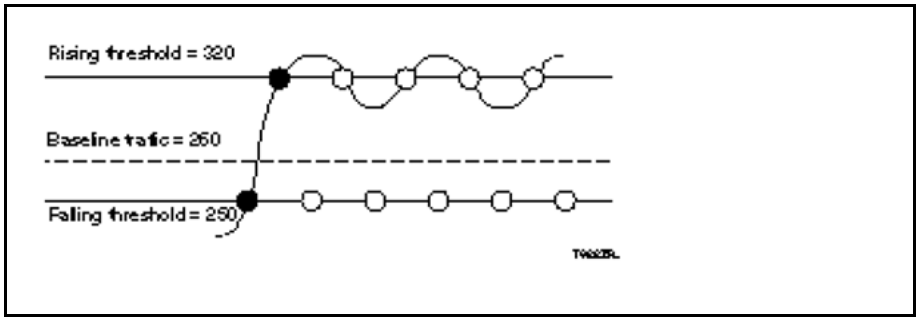
The alarm fires during the first interval that the sample goes out of range. No additional events generate for that threshold until the opposite threshold is crossed. Therefore, it is important you carefully define the rising and falling threshold values for alarms. Incorrect thresholds cause an alarm to fire at every alarm interval.

A general rule is to define one threshold value to an expected, baseline value, and then define the opposite threshold as the out-of-bounds limit. Because of sample averaging, the value is equal to ±1 baseline unit. For example, suppose you define an alarm with octets leaving a port as the variable. The intent of the alarm is to notify you if excessive traffic occurs on that port. You enable spanning tree, and then 52 octets transmit from the port every 2 seconds, which is equivalent to baseline traffic of 260 octets every 10 seconds. This alarm notifies you if the lower limit of exiting octets is defined at 260 and the upper limit is defined at 320 (or at any value greater than 260 + 52 = 312).

The first time outbound traffic other than spanning tree Bridge Protocol Data Units (BPDUs) occurs, the rising alarm fires. After outbound traffic other than spanning tree ceases, the falling alarm fires. This process provides the time intervals of any nonbaseline outbound traffic.

If you define the alarm with a falling threshold less than 260 (assuming the alarm polling interval is 10 seconds), for example, 250, then the rising alarm can fire only once, as shown in the following example. The falling alarm (the opposite threshold) must fire for the rising alarm to fire a second time. The falling alarm cannot fire unless the port becomes inactive or you disable spanning tree (which causes the value for outbound octets to drop to zero) because the baseline traffic is always greater than the value of the falling threshold. By definition, the failure of the falling alarm to fire prevents the rising alarm from firing a second time.

Figure 2
Alarm example, threshold less than 260



When you create an alarm, you select a variable from the variable list and a port, or another switch component to which it connects. Some variables require port IDs, card IDs, or other indexes (for example, spanning tree group IDs). You select a rising and a falling threshold value. The rising and falling values compare to the actual value of the variable that you choose. If the variable falls outside of the rising or falling value range, an alarm triggers and an event is logged or trapped.

When you create an alarm, you also select a sample type, which can be either absolute or delta. Define absolute alarms based on the cumulative value of the alarm variable. An example of an absolute alarm value is card operating status. Because this value is not cumulative, but instead represents states, such as card up (value 1) and card down (value 2), you configure it as the absolute value. Therefore, you can create an alarm with a rising value of 2 and a falling value of 1 to alert you whether the card is up or down.

Configure most alarm variables related to Ethernet traffic as a delta value. Define delta alarms based on the difference in the value of the alarm variable between the start of the polling period and the end of the polling period. Delta alarms are sampled twice for each polling period. For each sample, the last two values are added and compared to the threshold values. This process increases precision and detects threshold crossings that span the sampling boundary. Therefore, if you track the current values of a delta-valued alarm and add them, the result is twice the actual value. This result is not an error in the software.

RMON history

The RMON history group records periodic statistical samples from a network. A sample is a history and is gathered in time intervals referred to as buckets. You enable and create histories to establish a time-dependent method to gather RMON statistics on a port. The following are the default values for history:

- Buckets are gathered at 30-minute intervals.
- The number of buckets gathered is 50.

You can configure both the time interval and the number of buckets. However, after the last bucket is reached, bucket 1 is dumped and recycled to hold a new bucket of statistics. Then bucket 2 is dumped, and so forth.

RMON events

RMON events and alarms work together to notify you when values in your network go out of a specified range. After a value passes the specified range, the alarm fires. The event specifies how the activity is recorded.

An event specifies whether a trap, a log, or a trap and a log generates to view alarm activity. After you globally enable RMON, two default events generate:

- RisingEvent
- FallingEvent

The default events specify that after an alarm goes out of range, both a trap and a log track the firing of the alarm. For example, after an alarm fires at the rising threshold, the rising event specifies to send this information to both a trap and a log. Likewise, after an alarm passes the falling threshold, the falling event specifies to send this information to a trap and a log.

RMON statistics

You can use Device Manager to gather and graph Ethernet statistics in a variety of formats, or you can save them to a file and export them to a third-party presentation or graphing application.

This implementation of RMON requires a control row for Ethernet statistics. This control row appears as port 0/1 when you choose **RMON**, **Control**, **Ethernet Statistics**. The row ID is reserved for the control row. Therefore, some automated tests, such as ANVL, can fail when the test attempts to create a row 1.

Traps and logs

Use the information in this section to help you understand Simple Network Management Protocol (SNMP) traps and log files available as part of the Ethernet Routing Switch 8600 System Messaging Platform.

Simple Network Management Protocol

The Simple Network Management Protocol (SNMP) provides facilities for managing and monitoring network resources.

SNMP consists of

- agents—software running on a device that maintains information, about device configuration and current state, in a database
- managers—applications that contact an SNMP agent to query or modify the agent database
- the SNMP protocol—the application-layer protocol used by SNMP agents and managers to send and receive data
- Management Information Bases (MIB)—text files that specify the managed objects by an object identifier (OID).

ATTENTION
An Ethernet Routing Switch 8600 replies to SNMP requests to its physical IP address but not to SNMP requests to its VRRP virtual interface address.

An SNMP manager and agent communicate through the SNMP protocol. A manager sends queries, an agent responds and initiates traps.

There are several types of packets used between SNMP managers and agents:

- Get Request—requests the values of one or more objects
- Get Next Request—requests the value of the next object
- Set Request—requests modification of the value of one or more objects
- Get Response—message sent by an SNMP agent in response to a Get Request, Get Next Request, or Set Request message
- Trap—a notification triggered by events at the agent

Overview of traps and logs

The SNMP trap is an industry-standard method used to manage events. You can set SNMP traps for specific types of log messages (for example, WARNING, FATAL) from specific applications, and send them to a trap server for further processing. For example, you can configure the Ethernet Routing Switch 8600 to send SNMP traps to a server when a port is unplugged or when a power supply fails.

On any UNIX-based management platform, you can use system log (syslog) messaging to manage event messages. The Ethernet Routing Switch 8600 syslog software communicates with a server software component named *syslogd* on your management workstation.

The UNIX daemon syslogd is a software component that receives and locally logs, displays, prints, and forwards messages that originate from sources internal and external to the workstation. For example, syslogd on a UNIX workstation concurrently handles messages received from applications running on the workstation, as well as messages received from a Ethernet Routing Switch 8600 running in a network accessible to the workstation.

The remote UNIX management workstation does the following:

- receives system log messages from the Ethernet Routing Switch 8600
- examines the severity code in each message
- uses the severity code to determine appropriate system handling for each message

This document describes SNMP commands related to traps. For more information about configuring SNMP community strings and related topics, see *Nortel Ethernet Routing Switch 8600 Security* (NN46205-601).

System Messaging Platform

The System Messaging Platform (SMP) creates a scheme for the display and access of system messages. SMP enhances your access of information by offering greater serviceability. SMP helps in collecting, analyzing, and providing solutions to issues in a timely manner.

System Messaging Platform navigation

- "Log message format" (page 21)
- "Log files" (page 24)
- "Log file transfer" (page 27)

Log message format

The log messages for the Ethernet Routing Switch 8600 have a standardized format. All system messages are tagged with the following information:

- Module ID—software module from which the log is generated
- Nortel Proprietary (NP) information for debugging purposes.
- SF/CPU slot—identifies which slot of the SF/CPU generated the log message.
- Category—the category of the log message.
- Severity—the severity of the message.

The SMP message format is as follows:

```
<Module ID><Task><NP info><CPU slot><Time stamp><Category><Sev
erity>
```

The following is an example of an SMP message:

```
VLAN Task=tTrapd No-interface CPU5 [10/14/98 15:46:26] VLAN
WARNING Link Down
```

NP information is encrypted before it is written to the log file. The encrypted information is for debugging purposes. Only a Nortel Customer Service engineer can decrypt the information. The CLI commands display the logs without the encrypted information. Nortel recommends that you do not edit the log file.

The following table lists the system message categories.

Table 1
SMP categories

SMP categories			
ATM	IP	PIM	SNMP
CPU	IPMC	POLICY	STG
DVMRP	IP-RIP	POS	SW
EAP	IPX	QOS	VLAN
FILTER	MLT	RADIUS	WEB
HW	NONE	RIP	
IGMP	OSPF	RMON	

The following table describes the system message severity levels.

Table 2
SMP severity levels

Severity level	Definition
INFO	Information only. No action is required.
ERROR	A nonfatal condition occurred. You can be required to take appropriate action. For example, an error message is generated when the system is unable to lock onto the semaphore required to initialize the IP addresses used for transferring the SMP log file to a remote host.
WARNING	A nonfatal condition occurred. No immediate action is needed.
FATAL	A fatal condition occurred. The system cannot recover without rebooting. For example, a fatal message is generated when the configuration database is corrupted.

Based on the severity code in each message, the switch dispatches each message to any or all of the following destinations:

- workstation display
- local log file
- designated printer
- one or more remote hosts

Internally, the Ethernet Routing Switch 8600 has four severity levels for log messages: Info, Warning, Error, Fatal.

The system log supports eight different severity levels:

- Debug
- Info
- Notice
- Warning
- Critical
- Error
- Alert
- Emergency

The following table shows the default mapping of internal severity levels to syslog severity levels.

Table 3
Default and system log severity level mapping

UNIX system error codes	System log severity level	Internal Ethernet Routing Switch 8600 severity level
0	Emergency	Fatal
1	Alert	—
2	Critical	—
3	Error	Error
4	Warning	Warning
5	Notice	—
6	Info	Info
7	Debug	—

Log files

Log storage on the Ethernet Routing Switch 8600 is captured in two files:

- critical syslog file
- non-critical syslog file

The log file storage mechanism ensures that the system continues to log messages even if the PCMCIA or the compact flash card reaches its maximum storage limit, or if the attempt to send the log file to a remote server (FTP or TFTP) fails.

Nortel strongly recommends that you keep a PCMCIA card in each SF/CPU at all times.

Log file naming conventions

The log file is named according to 8.3 (*xxxxxxxx.sss*) format. The first six characters of the log file name contains the last three bytes of the chassis base MAC address. The next two characters specify the slot number of the SF/CPU that generated the logs. The last three characters (*sss*) denote the sequence number of the log file.

Critical syslog file

The critical syslog file contains all critical messages (all messages that have a severity level of ERROR or FATAL). Critical syslog files are stored on the PCMCIA card (the critical syslog file is never sent to the remote server). The critical syslog file has a fixed size of 500KB and all logs in the critical syslog file have a fixed size of 150 characters.

The critical syslog file uses a "first in, first out" (FIFO) mechanism (the newest message replaces the oldest message) to store critical log messages if the PCMCIA card reaches full capacity.

When you reboot the switch, logging of critical messages begins at the end of the log file with the most recent timestamp.

Crash dump information is not stored in the critical syslog file.

Example of critical syslog file

In this example, the critical syslog file contains log messages in the following order:

<NP> 24:ec25a43c2b85ddf8672920559c641f8bfbb5d3192bcdfe99</NP> CPU5 **[07/04/08 11:01:38]** SW INFO Closed telnet connection from 198.202.188.174, user rwa rcmd -2

<NP> 24:e00b73262be8dda921c7998c0da8c509e1678b41471f0cd5</NP> CPU5 **[07/04/08 11:08:19]** HW INFO Stand- by CPU in slot # 5 becoming master...

<NP> 24:2ff1e7b15c229788e686357f99951d2c209002f849c84183</NP> CPU5 **[07/04/08 11:00:42]** SW INFO CPU card entering warm-standby mode...

The system compares the timestamp of all consecutive log messages. In the preceding example, the timestamp of the first log message is less than the timestamp of the second log message, and the second log message has a timestamp greater than the timestamp of the third log message. In this case, the next log message that is recorded is placed after the second log message.

Non-critical syslog file

Non-critical syslog files contain all messages that have a severity level other than ERROR or FATAL. The system can generate multiple non-critical syslog files. If one syslog file reaches the maximum size limit and transfer to a remote server is not possible, then a new syslog file is created with an incremented sequence number. Log storage continues in the new non-critical syslog file. The sequence number of the log file identifies the version of the log file.

After a reboot of the switch, the system continues to log messages to the log file with the highest sequence number that is present in the PCMCIA. If no log file exists, then the system creates a new log file with a sequence number of “000” and the logs are stored in that file.

The system continues to log messages to the non-critical syslog file until it reaches maximum capacity, or until the PCMCIA card reaches its storage capacity. When the syslog file reaches maximum storage capacity, the file is transferred to the remote host you specify using FTP or TFTP. On successful transfer of the syslog file to the remote server, the syslog file is removed from the PCMCIA card and the system creates a new syslog file and increments the sequence number. Logging continues in the new file. Also, on successful transfer of the syslog file to the remote server, the system checks the PCMCIA card for all previous versions of non-critical syslog files and any such files are sent to the remote server (one at a time). An SNMP trap is generated for deletion of the syslog files.

If the transfer of the syslog file to the remote server fails or if the remote server is unreachable (that is, the file is not transferred to the remote server), then the syslog file is not deleted. An SNMP trap is generated when transfer to the FTP or TFTP server fails. A new non-critical syslog file with an incremented sequence number is created and logging continues in the new syslog file. The system creates a new syslog file for

every FTP or TFTP failure until the storage capacity of the PCMCIA card is reached. When no free space remains on the PCMCIA card, the system deletes the oldest syslog file and creates a new file.

Before logging a system message on the PCMCIA card, SMP calculates the space available for logging according to the parameters defined. If there is insufficient storage capacity on the PCMCIA card for one syslog file or for more than one, the system generates an error message to alert you.

ATTENTION
Make sure you have sufficient space for the SMP log on your PCMCIA card. Smaller amounts of free space for the log cause more frequent transfers.

Example of non-critical syslog file

Two syslog files are present in the PCMCIA (xxxxxxxx.004 and xxxxxxxx.005). Logs are stored in the .005 file. When the .005 file reaches maximum capacity, the system attempts to send the log file to a remote server (FTP or TFTP). The system processes the files using the following actions:

1. If transfer of the .005 syslog file is successful, then a new syslog file is created (xxxxxxxx.006) and the .005 file is deleted from the PCMCIA card. The PCMCIA is traced for all files with a sequence number less than .005. If any such files are found, the system also sends those to the remote server. In this example, the .004 file is also sent to the FTP or TFTP server.
   - If file .004 transfers successfully, the file is deleted from the PCMCIA card.
   - If the transfer of file .004 is unsuccessful, it is not deleted from the PCMCIA card.
2. If transfer of the .005 syslog file fails, the system checks the PCMCIA card for additional storage space.
   - If there is additional storage space available, then the .006 syslog file is created.
   - If the PCMCIA card has reached maximum storage capacity, then the .004 file is deleted and the new syslog file (version .006) is created.
3. The preceding two steps are followed for subsequent logs. If, at any point in time, there is only one file stored in the PCMCIA card, and there is no additional space available on the card, then an error message is generated.

Log file transfer

The system logs contain important information for debugging and maintaining your Ethernet Routing Switch 8600. When logging to the PCMCIA card, the log file is automatically transferred to a remote host when it reaches your specified size parameters. You can configure up to 10 remote hosts, creating long-term backup storage of your system log files.

Of the 10 configured remote hosts, 1 is the primary host and the other 9 are redundant. Upon initiating a transfer, SMP always attempts to use host 1 first. If host 1 is not reachable, SMP tries host 2, and then host 3, and so on.

You can specify the following information to configure the transfer criteria:

- Configurable log size parameters for the PCMCIA include:
  - minsize—the minimum acceptable free space available on the PCMCIA for logging
  - maxsize—the maximum size of the log file on the PCMCIA
  - maxoccupyPercentage—the amount of memory to use for SMP logging when the maxsize parameter cannot be met
- The IP address of the remote host.
- The name of the log file that is to be stored on the remote host.
- The user name and password, if required. You can use the following command to configure the user name and password:
  config bootconfig host user <value> password <value>

Be aware of the following restrictions when transferring log files to a remote host:

- The remote host IP address must be reachable.
- When you transfer a log file from a host to the switch, (for example, to display it with the show log file command), you should rename the log file. Failure to rename the log file can cause the switch to use the recently transferred file as the current log, if the sequence number in the extension is higher than the current log file. For example, if bf860005.002 is the current log file and you transfer bf860005.007 to the switch, the switch logs future messages to the bf860005.007 file. You can avoid this if you rename the log file to something other than the format used by SMP.
- If your TFTP server is a UNIX-based machine, any files written to the server must already exist. For example, you must create dummy files with the same names as your system logs. This is commonly done by using the touch command (for example, **touch bf860005.001**).

Link state change control

Rapid fluctuation in a port link state is called link flapping.

Link flapping is detrimental to network stability because it can trigger recalculation in spanning tree and the routing table.

If the number of port down events exceeds a configured limit during a specified interval, the system forces the port out of service.

You can configure link flap detection to control link state changes on a physical port. When you configure link flap detection, you can set thresholds for the number and frequency of changes allowed.

You can configure the system to take one of the following actions if changes exceed the thresholds:

- send a trap
- bring down the port

If changes exceed the link state change thresholds, the system generates a log entry.

RMON configuration using Enterprise Device Manager

Remote monitoring (RMON) is a management information base (MIB) or a group of management objects that you use to obtain or configure values using the Simple Network Management Protocol (SNMP).

Navigation

- "Enabling RMON globally" (page 29)
- "Enabling RMON history" (page 31)
- "Disabling RMON history" (page 33)
- "Creating alarms" (page 34)
- "Viewing RMON alarms" (page 36)
- "Viewing RMON events" (page 39)
- "Viewing the RMON log" (page 40)
- "Deleting alarms" (page 40)
- "Creating RMON events (default)" (page 41)
- "Creating events (nondefault)" (page 42)
- "Deleting events" (page 43)
- "Disabling RMON statistics" (page 43)

Enabling RMON globally

You must globally enable RMON before you use an RMON function. If you attempt to enable any RMON function when the global flag is disabled, Enterprise Device Manager informs you that the flag is disabled and prompts you to enable the flag.

Globally enable RMON by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**. If you want to use nondefault RMON parameter values, you can configure them before you enable RMON or as you configure the RMON functions.
2	Double-click **Options.**
3	Select a utilization method.
4	Select a trap option.
5	Select a memory size.
6	Select **Enable** to enable RMON.
7	Click **Apply**.

--End--

Variable definitions

Use the data in the following table to use the RmonOptions, Options fields.

Variable	Value
Enable	Enables RMON. If you select the Enable box, the RMON agent starts immediately if the amount of memory specified by MemSize is currently available in the device. To disable RMON, clear the Enable box, click **Apply** to save the new setting to NVRAM, and restart the device. The default is disabled.
UtilizationMethod	Controls whether RMON uses a half-duplex or full-duplex formula to calculate port usage. When you select halfDuplex, RMON uses InOctets and the speed of the port to calculate port usage (this is the standard RMON rfc1271 convention). When you select fullDuplex, RMON uses InOctets and OutOctets and 2X the speed of the port to calculate port usage. If you select fullDuplex, but the port operates in half-duplex mode, the calculation defaults to the rfc1271 convention. The default is halfDuplex.

Variable	Value
TrapOption	Indicates whether the system sends RMON traps to the owner of the RMON alarm (the manager that created the alarm entry) or to all trap recipients in the system trap receiver table. The default value is toOwner.
MemSize	Specifies the RAM size, in bytes, available for RMON to use. The default value is 250 Kilobytes.

Enabling RMON history

Use RMON to establish a history for a port and configure the bucket interval. For example, to gather RMON statistics over the weekend, you must have enough buckets to cover two days. Configure the history to gather one bucket every hour, and cover a 48 hour period. After you configure history characteristics, you cannot modify them; you must delete the history and create another one.

Use this procedure to enable RMON history.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**.
2	Double-click **Control.**
3	Click **Insert**.
4	In the **Port** box, click the ellipsis button to select a port.
5	In the **Buckets Requested** box, enter the number of discrete time intervals to save data.
6	Enter the **Interval** in seconds.
7	In the **Owner** box, enter owner information.
8	Click **Insert**.

--End--

Variable definitions

Use the data in the following table to view RMON history fields.

Variable	Value
Index	Specifies index that uniquely identifies an entry in the historyControl table. Each entry defines a set of samples at a particular interval for an interface on the device. Index value ranges from 1–65535. The default value is 1.
Port	Identifies the source for which historical data is collected and placed in a media-specific table on behalf of this historyControlEntry. The source is an interface on this device. To identify a particular interface, the object identifies the instance of the ifIndex object, defined in [4,6], for the desired interface. For example, if an entry receives data from interface 1, the object is ifIndex.1. The statistics in this group reflect all packets on the local network segment attached to the identified interface. You cannot modify this object if the associated historyControlStatus object is equal to valid(1).
BucketsRequested	Specifies the requested number of discrete time intervals over which data is saved in the part of the media-specific table associated with this historyControlEntry. When this object is created or modified, the probe configures historyControlBucketsGranted as closely to this object as possible for the particular probe implementation and available resources. Values range from 1–65535. The default value is 50.
BucketsGranted	Specifies the number of discrete sampling intervals over which data is saved in the part of the media-specific table associated with this historyControlEntry. When the associated BucketsRequested object is created or modified, the probe sets this object as closely to the requested value as possible for the particular probe implementation and available resources. The probe must not lower this value except as a result of a modification to the associated BucketsRequested object. Occasionally, the actual number of buckets associated with this entry is less than the value of this object. In this case, at the end of each sampling interval, a new bucket is added to the media-specific table. When the number of buckets reaches the value of this object and a new bucket is to be added to the media-specific table, the oldest bucket associated with this entry is deleted by the agent so that the new bucket can be added. When the value of this object changes to a value less than the current

Variable	Value
	value, entries are deleted from the media-specific table associated with this entry. The agent deletes the oldest of these entries so that their number remains less than or equal to the new value of this object. When the value of this object changes to a value greater than the current value, the number of associated media-specific entries is allowed to grow.
Interval	Specifies the interval in seconds over which the data is sampled for each bucket in the part of the media-specific table associated with this historyControlEntry. You can set this interval to any number of seconds from 1–3600 (1 hour). Because the counters in a bucket can overflow at their maximum value with no indication, a prudent manager takes into account the possibility of overflow in any of the associated counters. Consider the minimum time in which any counter can overflow on a particular media type and set the historyControlInterval object to a value less than this interval. This is typically most important for the octets counter in a media-specific table. For example, on an Ethernet network, the etherHistoryOctets counter can overflow in about 1 hour at the maximum utilization. You cannot modify this object if the associated historyControlStatus object is equal to valid. The default value is 1800.
Owner	Specifies the entity that configured this entry and is using the assigned resources.

Disabling RMON history

Disable RMON history on a port when you do not want to record a statistical sample from that port.

Disable RMON history by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**.
2	Double-click **Control**.
3	Select the row that contains the port ID to delete.

4 Click **Delete**.

--End--

Creating alarms

Ensure that RMON is globally enabled. When you enable RMON globally, you also create a default rising and falling event. The default for the events is log-and-trap, which means that you receive notification through a trap as well as through a log file.

A list of variable definitions is in “RMON alarm variables” (page 129).

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON** .
2	Double-click **Alarms**.
3	In the **Variable** menu on the Alarm Manager dialog box, select a variable for the alarm. Depending on the variable you select, you are prompted for a port (or other object) on which you want to set an alarm. Alarm variables exist in three formats, depending on the type: • A chassis, power supply, or fan-related alarm ends in x where the x index is hard-coded. No further information is required. • A card, spanning tree group (STG), Routing Information Protocol (RIP) or Open Shortest Path First (OSPF), or EtherStat alarm ends with a dot (.). You must enter a card number, STG ID, IP address, or EtherStat information. • A port alarm ends with no dot or index and requires using the port shortcut menu. An example of a port alarm is ifInOctets (interface incoming octet count).
4	Select a sample type.
5	Type a sample interval in seconds.
6	Type a number in the **Index** field.
7	In the **Threshold Type** section, enter rising and falling values.
8	Click **Insert**.

--End--

Variable definitions

Use the data in the following table to use the Alarm Manager dialog box fields.

Variable	Value
Variable	Specifies the name and type of alarm—indicated by the format • *alarmname.x*, where x=0 indicates a chassis alarm, x=1 or 2 indicates a power supply or fan alarm with 1 being the primary unit and 2 the secondary unit. • *alarmname*, where the user must specify the index. This value is a card number for module-related alarms, an STG ID for spanning tree group alarms (the default STG is 1; other STG IDs are user configured), an IP address for RIP or OSPF alarms (RIP/OSPF must be enabled on the VLAN or router port and enabled globally), or the Ether Statistics Control Index for RMON Stats alarms. • *alarmname* with no dot or index is a port-related alarm and results in display of the port picker tool.
SampleType	Specifies the sample type. Value can be absolute or delta. Default value is delta.
Sample Interval	Specifies the Time period (in seconds) over which the data is sampled and compared with the rising and falling thresholds. Default value is 10 seconds.
Index	Uniquely identifies an entry in the alarm table. Each such entry defines a diagnostic sample at a particular interval for an object on the device. The default value is 1.
Threshold type	• Rising Value: Index of the event entry that is used when a rising threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.) • Falling Value: Index of the event entry that is used when a falling threshold is crossed. The event entry identified by a particular value of this index is the same as identified by the same value of the event index object. (Generally, accept the default that is already filled in.)

Variable	Value
Value	• Rising value: When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval was less than this threshold, generates a single event. • Falling value: When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval was greater than this threshold, generates a single event.
Event Index	Index of the event entry that is used when a rising threshold is crossed. Index of the event entry that is used when a falling threshold is crossed.

Viewing RMON alarms

View the RMON alarm information to see alarm activity by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**.
2	Double-click **Alarms**.

--End--

Variable definitions

Use the following table to use the RmonAlarms, Alarms fields.

Variable	Value
Index	Uniquely identifies an entry in the alarm table. Each entry defines a diagnostic sample at a particular interval for an object on the device.
Interval	Specifies the interval, in seconds, over which the data is sampled and compared with the rising and falling thresholds. deltaValue sampling—configure the interval short enough that the sampled variable is unlikely to increase or decrease by more than 2^31–1 during a single sampling interval.

Variable	Value
Variable	Specifies the object identifier of the particular variable to be sampled. Only variables that resolve to an ASN.1 primitive type of INTEGER (INTEGER, Counter, Gauge, or TimeTicks) can be sampled. Because SNMP access control is articulated entirely in terms of the contents of MIB views, no access control mechanism exists to restrict the value of this object to identify only those objects that exist in a particular MIB view. Because no acceptable means of restricting the read access that is obtained through the alarm mechanism exists, the probe must grant only write access to this object in those views that have read access to all objects on the probe. During a set operation, if the supplied variable name is not available in the selected MIB view, a badValue error must be returned. If at any time the variable name of an established alarmEntry is no longer available in the selected MIB view, the probe must change the status of this alarmEntry to invalid. You cannot modify this object if the associated alarmStatus object is equal to valid.
SampleType	Specifies the method of sampling the selected variable and calculating the value to be compared against the thresholds. If the value of this object is absoluteValue, the value of the selected variable is compared directly with the thresholds at the end of the sampling interval. If the value of this object is deltaValue, the value of the selected variable at the last sample is subtracted from the current value, and the difference compared with the thresholds. You cannot modify this object if the associated alarmStatus object is equal to valid.
Value	Specifies the value of the statistic during the last sampling period. For example, if the sample type is deltaValue, this value is the difference between the samples at the beginning and end of the period. If the sample type is absoluteValue, this value is the sampled value at the end of the period. This is the value that is compared with the rising and falling thresholds. The value during the current sampling period is not made available until the period is completed and remains available until the next period is complete.

Variable	Value
StartUpAlarm	Specifies the alarm that is sent when this entry is first set to valid. If the first sample after this entry becomes valid is greater than or equal to the risingThreshold and alarmStartupAlarm is equal to risingAlarm or risingOrFallingAlarm, and then a single rising alarm is generated. If the first sample after this entry becomes valid is less than or equal to the fallingThreshold and alarmStartupAlarm is equal to fallingAlarm or risingOrFallingAlarm, and then a single falling alarm is generated. You cannot modify this object if the associated alarmStatus object is equal to valid.
Rising Threshold	Specifies a threshold for the sampled statistic. When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval was less than this threshold, a single event is generated. A single event is also generated if the first sample after this entry becomes valid is greater than or equal to this threshold and the associated alarmStartupAlarm is equal to risingAlarm or risingOrFallingAlarm. After a rising event is generated, another such event is not generated until the sampled value falls below this threshold and reaches the alarmFallingThreshold. You cannot modify this object if the associated alarmStatus object is equal to valid.
RisingEventIndex	Specifies the index of the eventEntry that is used when a rising threshold is crossed. The eventEntry identified by a particular value of this index is the same as identified by the same value of the eventIndex object. If there is no corresponding entry in the eventTable, no association exists. In particular, if this value is zero, no associated event is generated, as zero is not a valid event index. You cannot modify this object if the associated alarmStatus object is equal to valid.
FallingThreshold	Specifies a threshold for the sampled statistic. When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval was greater than this threshold, a single event is generated. A single event is also generated if the first sample after this entry becomes valid is less than or equal to this threshold and the associated alarmStartupAlarm is equal to fallingAlarm or risingOrFallingAlarm. After a falling event is generated, another such event is not generated until the sampled value rises above this threshold and reaches the alarmRisingThreshold. You cannot modify this object if the associated alarmStatus object is equal to valid.

Variable	Value
FallingEventIndex	Specifies the index of the eventEntry that is used when a falling threshold is crossed. The eventEntry identified by a particular value of this index is the same as identified by the same value of the eventIndex object. If there is no corresponding entry in the eventTable, no association exists. In particular, if this value is zero, no associated event is generated, as zero is not a valid event index. You cannot modify this object if the associated alarmStatus object is equal to valid.
Owner	Specifies the entity that configured this entry and is therefore using the resources assigned to it.
Status	Specifies the status of this alarm entry.

Viewing RMON events

View RMON events to see how many events occurred by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**.
2	Double-click **Alarms**.
3	Click the **Events** tab.

--End--

Variable definitions

Use the data in the following table to use the Events tab.

Variable	Value
Index	Uniquely identifies an entry in the event table. Each such entry defines one event that is generated when the appropriate conditions occur.
Description	Specifies a comment describing this event entry.
Type	Specifies the type of notification that the probe makes about this event. In the case of a log, an entry is made in the log table for each event. In the case of snmp-trap, an SNMP trap is sent to one or more management stations.
Community	If an SNMP trap is to be sent, it is sent to the SNMP community specified by this octet string.

Variable	Value
LastTimeSent	Specifies the value of sysUpTime at the time this event entry last generated an event. If this entry has not generated any events, this value is zero.
Owner	Specifies the entity that configured this entry and is therefore using the assigned resources. If this object contains a string starting with monitor and has associated entries in the log table, all connected management stations retrieve those log entries, as they have significance to all management stations connected to this device.

Viewing the RMON log

View the Trap log and see which activity occurred by using the bell icon on the Device Manager toolbar.
View RMON log by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**.
2	Double-click **Alarms**.
3	Click the **Log** tab. The RmonAlarms—Log tab appears showing log information.

--End--

Variable definitions

Use the data in the following table to use the Log tab.

Variable	Value
Time	Specifies the creation time for this log entry.
Description	Specifies an implementation dependent description of the event that activated this log entry.

Deleting alarms

Delete an alarm when you no longer want it to appear in the log by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Serviceability, RMON**.
2	Double-click **Alarms**.
3	Select the alarm to delete.
4	Click **Delete**.

--End--

Creating RMON events (default)

Create a default rising and falling event to specify when alarm information is sent to a trap and a log by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Servicability, RMON.**.
2	Double-click **Alarms**.
3	Click the **Events** tab.
4	Click **Insert**.
5	In the Insert Events dialog box, enter a value in the **Index** field.
6	Enter a comment describing this event in the **Description** field.
7	Select the type of notification besides the **Type** field.
8	Enter the type of community in the **Community** field.
9	Enter the owner in the **Owner** field.
10	Click **Insert**. If Rmon is not globally enabled, the following message appears: *RMON is currently disabled. Do you want to enable it now?*
11	Click **Yes**.

--End--

For more information, see “Variable definitions” (page 41).

Variable definitions

Use the data in the following table to use the Events tab.

Variable	Value
Index	Uniquely identifies an entry in the event table. Each such entry defines one event that is generated when the appropriate conditions occur.
Description	Specifies a comment describing this event entry.
Type	Specifies the type of notification that the probe makes about this event. In the case of a log, an entry is made in the log table for each event. In the case of snmp-trap, an SNMP trap is sent to one or more management stations.
Community	If an SNMP trap is to be sent, it is sent to the SNMP community specified by this octet string.
Owner	Specifies the entity that configured this entry and is therefore using the assigned resources. If this object contains a string starting with monitor and has associated entries in the log table, all connected management stations retrieve those log entries, as they have significance to all management stations connected to this device.

Creating events (nondefault)

Create a custom rising and falling event to specify when alarm information is sent to a trap, a log, or a trap and a log by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Serviceability, RMON.**.
2	Double-click **Alarms**.
3	Click the **Events** tab.
4	Click **Insert**.
5	Type an event name in the **Description** field of the RmonAlarms, Insert Events dialog box.
6	Select the type of event you want. The default setting is log-and-trap. To save memory, set the event type to log. To reduce traffic from the switch, set the event type to snmp-log. If you select snmp-trap or log, you must set trap receivers.
7	Click **Insert**.

The new event appears in the Events tab of the RmonAlarms dialog box.

--End--

Deleting events

Delete an event when you no longer require the alarm information by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Serviceability, RMON.**.
2	Double-click **Alarms**.
3	Click the **Events** tab.
4	Select the event to delete.
5	Click **Delete**.

--End--

Disabling RMON statistics

Disable RMON statistics on a port when you do not want to gather statistics on that port by performing this procedure.

Procedure steps

Step	Action
1	In the navigation pane, open the following folders:**Serviceability, RMON**.
2	Double-click **Control**.
3	Click the **Ethernet Statistics** tab.
4	Select the row that contains the port ID for which you want to disable statistics.
5	Click **Delete**.

--End--

RMON configuration using the CLI

This chapter contains procedures to configure Remote Monitoring (RMON) on the Nortel Ethernet Routing Switch 8600 by using the command line interface (CLI).

Navigation

- "Configuring RMON" (page 46)
- "Viewing RMON Settings" (page 49)
- "Configuring the switch to capture RMON statistics" (page 50)

Job aid: Roadmap of CLI commands for configuring RMON

The following table describes commands and parameters to configure RMON.

Command	Parameter
config rmon	alarm create <id> type <value> intv <value> [variable <value>] [r_th <value>] [r_ev <value>] [f_th<value>] [f_ev <value>] [owner <value>]
	alarm delete <id>
	alarm info
	disable
	enable
	ether-stats create <id> <ports> [owner <value>]
	ether-stats delete <id>
	ether-stats info
	ether-stats owner <id> <name>
	event create <id> [desc <value>] [type <value>] [community <value>] [owner <value>] [trap_src <value>] [trap_dest<value>]
	event delete <id>

Command	Parameter
	event info
	history-control create <id> <ports> [buckets <value>] [intv <value>] [owner <value>]
	history-control delete <id>
	history-control info
	memsize <memsize>
	info
	trap-option <toOwner|toAll>
	util-method <half|duplex>
show rmon	alarm
	ether-stats
	event
	history-control
	info
	log
	show-all [file <value>]
monitor ports stats rmon	[<ports>] [from <value>]

Configuring RMON

Configure RMON functions on the switch to set alarms and capture events by performing this procedure.

Procedure steps

Step	Action
1	Configure RMON functions on the switch: **config rmon**

--End--

Variable definitions

The following table describes variables that you enter after the **config rmon** command.

Variable	Value
alarm create <id> type <value> intv <value> [variable <value>] [r_th <value>] [r_ev <value>] [f_th<value>] [f_ev <value>] [owner <value>]	Creates an alarm interface. • **id** is the interface index number (1–65535). • **type <value>** is the sample type, **absolute** or **delta**. • **intv <value>** is the sample interval (1–3600). • **variable <value>** is the variable name or object identifier (OID), case sensitive (string length 1–1536). • **r_th <value>** is the rising threshold (-2147483647 to 2147483647). • **r_ev <value>** is the rising event number (1–65535). • **f_th <value>** is the falling threshold (-2147483647 to 2147483647). • **f_ev <value>** is the falling event number (1–65535). • **owner <value>** is the name of the owner (string length 1–127).
alarm delete <id>	Deletes the specified RMON alarm index number expressed as a value from 1–65535.
alarm info	Displays information about the RMON alarms.
disable	Disables RMON on the switch.
enable	Enables RMON on the switch.
ether-stats create <id> <ports> [owner <value>]	Creates an ether-stats control interface. • **id** is the index number of the ether stats control interface (1–65535). • **ports** is the single port interface {slot/port[-slot/port][,...]}. • **owner <value>** is name of the owner (string length 1–127).
ether-stats delete <id>	Deletes an ether-stats control interface. **id** is the index number of the ether stats control interface (1–65535).
ether-stats info	Displays the current ether-stats settings.

Variable	Value
ether-stats owner <id> <name>	Changes the owner name for the ether-stats control interface. • **id** is the index number of the ether stats control interface (1–65535). • **name** is name of the owner (string length 1–127).
event create <id>] [trap_src <value>] [tra p_dest<value>] [desc <value>] [type <value>] [community <value>] [owner <value>	Creates an event. • **id** is the event index number (1–65535). • **desc <value>** is the event description (string length 0–127). • **type <value>** is the event type, none, log, snmp-trap, or log-and-trap. • **community <value>** is the event community (string length 1–127). • **owner <value>** is the name of the owner (string length 1–127). • **trap_src <value>** is the trap source ip address. • **trap_dest <value>** is the trap destination ip address.
event delete <id>	Deletes an event. **id** is the event index number (1–65535).
event info	Displays the event information.
history-control create <id> <ports> [buckets <value>] [intv <value>] [owner <value>]	Creates a history control interface. • **id** is the index number of the history control interface (1–65535). • **ports** is the single port interface {slot/port[-slot/p ort][,...]}. • **buckets <value>** is the number of buckets requested (1–65535). • **intv <value>** is the time interval in seconds over which the data is sampled for each bucket (1–3600). • **[owner <value>** is the name of the owner (string length 1–127).
history-control delete <id>	Deletes a history control interface. **id** is the alarm index number (1–65535).
history-control info	Displays the setting for history control interfaces.

Variable	Value
memsize <memsize>	Configures the amount of RAM in bytes to allocate for RMON. **memsize** is the memory size in bytes (250000–4000000).
info	Indicates whether RMON is enabled or disabled on the switch.
trap-option <toOwner|toAll>	Controls whether the RMON traps are sent to the owner or to all trap recipients. **toOwner|toAll** is configures as either the owner or to all trap recipients.
util-method <half|duplex>	Controls whether port utilization is calculated in half or full duplex.

Example of configuring RMON

Procedure steps

Step	Action
1	Enable RMON: **8610:5/config/rmon# enable**

2	Display information about RMON:
	8610:5/config/rmon# info
	`Sub-Context: alarm ether-stats event history-control`
	`Current Context:`
	`rmon : enable`
	`mansize : 250000`
	`trap-option : toOwner`

--End--

Viewing RMON Settings

View RMON settings to see information about alarms, statistics, events, or the status of RMON on the switch by performing this procedure.

Procedure steps

Step	Action
1	View RMON settings: **show rmon**

--End--

Job aid: Output for show rmon

The following table describes the output for the **show rmon** command.

Parameter	Description
alarm	Displays the RMON Alarm table.
ether-stats	Displays the RMON Ethernet statistics table.
event	Displays the RMON event table.
history-control	Displays the RMON history control table.
info	Displays the status of RMON on the switch.
log	Displays the RMON log table.
show-all [file <value>]	Displays all RMON information. • **file <value>** is the file name, /pcmcia/<file> | /flash/<file> expressed as a string from 1–99 of characters.

Configuring the switch to capture RMON statistics

Configure the switch to capture RMON statistics to monitor network performance by performing this procedure.

Procedure steps

Step	Action
1	Configure the switch to capture RMON statistics: **monitor ports stats rmon [<ports>] [from <value>]**
	--End--

See the following table for more information.

Variable definitions

Use the following table to complete the **monitor ports stats rmon** command.

Variable	Value
[<ports>]	Indicates the ports on which you want to capture statistics.
[from <value>]	Indicates a range of ports.

RMON configuration using the NNCLI

This chapter contains procedures to configure Remote Monitoring (RMON) on the Nortel Ethernet Routing Switch 8600 by using the Nortel Networks command line interface (NNCLI).

Navigation

- "Configuring RMON" (page 53)
- "Viewing RMON settings" (page 56)

Job aid: Roadmap of RMON commands

The following table lists the commands and parameters that you use to perform the procedures in this section.

Command	Parameter
Privileged EXEC mode	
`monitor ports statistics rmon`	`[<portList>]` `[from <portList>]`
`show rmon`	—
`show rmon alarm`	—
`show rmon event`	—
`show rmon history`	—
`show rmon log`	—
`show rmon stats`	—
Global Configuration mode	
`default rmon`	—
`default rmon alarm`	`<1-65535> [owner]`

Command	Parameter
default rmon event	<1-65535> [owner] [community] [description]
default rmon history	<1-65535> [buckets] [interval] [owner]
default rmon memsize	—
default rmon stats	<1-65535> [owner]
default rmon trap-option	—
default rmon util-method	—
no rmon	—
no rmon alarm	[<1-65535>]
no rmon event	[<1-65535>] [log]
no rmon history	[<1-65535>]
no rmon stats	[<1-65535>]
rmon	—
rmon alarm	<1-65535> <OID:LINE> <1-2147483647> {absolute | delta} rising-threshold <-2147483648-2147483647> [<event:1-65535>] falling-threshold <-2147483648-2147483647> [<event:1-65535>] [owner <LINE>]

Command	Parameter
rmon event	<1-65535> [log] [trap] [description <LINE>] [owner <LINE>] [trap_src <A.B.C.D>] [trap_dest <A.B.C.D>] [community <WORD/1-127>]
rmon history	<1-65535> <portList> [<buckets:1-65535>] [<interval:1-3600>] [owner <LINE>]
rmon memsize	<250000-4000000>
rmon stats	<1-65535> <portList> [owner <LINE>]
rmon trap-option	<toOwner|toAll>
rmon util-method	<half|full>

Configuring RMON

Configure RMON functions on the switch to set alarms and capture events by performing this procedure.

Prerequisites

- You must log on to Global Configuration mode.

Procedure steps

Step	Action
1	Enable RMON globally: **rmon**
2	Configure RMON alarms on the switch: **rmon alarm**
3	Configure RMON events on the switch: **rmon event**

--End--

Variable definitions

Use the data in the following table to use the **rmon** command.

Variable	Value
alarm <1-65535> <Word/1-536> <1-3600> {**absolute** | **delta**} **rising-threshold** <-2147483648-2147483647> [<event:1-65535>] **falling-threshold** <-2147483648-2147483647> [<event:1-65535>] [**owner** <WORD/2-127>] **default rmon alarm** <1-65535>	Creates an alarm interface. • <1-65535> is the interface index number from 1–65535. • <Word/1-1536> is the variable name or object identifier (OID), case sensitive (string length 1–1536). • <1-3600> is the sample interval, which is the time period (in seconds) over which the data is sampled and compared with the rising and falling thresholds. Default value is 10 seconds. • {**absolute** | **delta**} is the sample type. • **rising-threshold** <-2147483648-2147483647> [<event:1-65535>] is the rising threshold (–2147483648–2147483647) and the rising event number (1–65535). • **falling-threshold** <-2147483648-2147483647> [<event:1-65535>] is the falling threshold (–2147483648–2147483647) and the falling event number (1–65535). • **owner** <WORD/2-127> is the name of the owner (string length 1–48). Use the **default rmon alarm** <65535> command to configure the default RMON alarm.

Variable	Value
	Use the **no** operator to disable RMON alarms: **no rmon alarm [<1-65535>]**
stats <1-65535> <portList> [owner <WORD/1-127>]	Creates an ether-stats control interface. • **<1-65535>** is the index number of the ether stats control interface. • **portList** is the single port interface {slot/port[-slot/port][,...]}. • **owner <WORD/1-127>** is name of the owner (string length 1–127). Use the **no** operator to delete a stats control interface: **no rmon stats [<1-65535>]**
event <1-65535> [log] [trap] [description <LINE>] [owner <LINE>] [trap_src <A.B.C.D>] [trap_dest <A.B.C.D>] [community <WORD/1-127>]	Creates an event. • **<1-65535>** is the event index number. • **[log]** displays information about configured traps. • **[trap]** specifies trap source and destination IP addresses. • **description <LINE>** is the event description (string length 0–127). • **owner <WORD/1-127>** is the name of the owner (string length 1–127). • **trap_src <A.B.C.D>** is the trap source ip address. • **trap_dest <A.B.C.D>** is the trap destination ip address. • **community <WORD/1-27>** is the event community (string length 1–127). Use the **no** operator to delete a RMON event: **no rmon event [<1-65535>] [log]**

Variable	Value
history <1-65535> <portList> [<buckets:1-65535>] [<interval:1-3600>] [owner <WORD/1-127>]	Creates a history control interface. • **<1-65535>** is the index number of the history control interface (1–65535). • **<portList>** is the single port interface {slot/port[-slot/port][,...]}. • **[<buckets:1-65535>]** is the number of buckets requested (1–65535). • **[<interval:1-3600>]** is the time interval in seconds over which the data is sampled for each bucket (1–3600). • **[owner <WORD/1-127>]** is the name of the owner (string length 1–48). Use the **no** operator to delete a history control interface: **no rmon history [<1-65535>]**
memsize <250000-4000000>	Configures the amount of RAM in bytes to allocate for RMON. The range is 250000–4000000.
trap-option <toOwner | toAll>	Controls whether the RMON traps are sent to the owner or to all trap recipients. **toOwner | toAll** is set to either the owner or to all trap recipients.
util-method <half | full>	Controls whether port utilization is calculated in half or full duplex.

Viewing RMON settings

View RMON settings to see information about alarms, statistics, events, or the status of RMON on the switch by performing this procedure.

Prerequisites

- You must log on to Privileged EXEC mode.

Procedure steps

Step	Action
1	View RMON settings:

```
show rmon
```

```
--End--
```

Job aid: Output for show rmon

Use the data in the following table to use the **show rmon** command.

Parameter	Description
alarm	Displays the RMON Alarm table.
event	Displays the RMON event table.
history	Displays the RMON history table.
log	Displays the RMON log table.
stats	Displays the RMON statistics table.

Log and trap configuration using Enterprise Device Manager

Use logs and traps as part of fault management operations and to provide diagnostic information in troubleshooting procedures.

Navigation

- "SNMP trap configuration" (page 59)
- "Log configuration" (page 67)
- "Viewing Enterprise Device Manager logs" (page 70)

SNMP trap configuration

Use Simple Network Management Protocol (SNMP) traps and notifications to allow management stations to gather information about switch activities, alarms, and other information.

Configure traps by creating SNMP trap notifications, creating a target address to which you want to send the notifications, and specifying target parameters.

Specify which protocols and processes generate traps by enabling traps for that protocol. For example, to allow SNMP traps to be generated for Open Shortest Path First (OSPF), use the following command: **`config ip ospf trap enable`**.

For more information about configuring SNMP community strings and related topics, see *Nortel Ethernet Routing Switch 8600 Security* (NN46205-601).

Navigation

- "Configuring an SNMP host target address" (page 60)
- "Configuring target table parameters" (page 61)
- "Viewing the trap sender table" (page 62)

- "Configuring an SNMP notify table" (page 63)
- "Configuring SNMP notify filter profile table parameters" (page 64)
- "Configuring SNMP notify filter table parameters" (page 65)
- "Viewing SNMP trap logs" (page 66)

Configuring an SNMP host target address

If you are using an SMMPv3-enabled switch, use this procedure to configure a target table to specify the list of transport addresses to use in the generation of SNMP messages.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Edit, SnmpV3.**.
2	Double-click **Target Table**.
3	Click **Insert**.
4	Type a unique identifier in the **Name** box.
5	Type the transport type of the address in the **TDomain** box.
6	Type the transport address in the **TAddress** box..
7	Type the maximum round trip time in the **Timeout** box.
8	Type the number of retries to be attempted in the **RetryCount** box.
9	Type the list of tag values in the **TagList** box.
10	Type the SnmpAdminString in the **Params** box.
11	Type the mask in the **TMask** box.
12	Type the maximum message size in the **MMS** box.
13	Click **Insert**.

--End--

Variable definitions

Use the data in the following table to configure a target table.

Variable	Value
Name	Specifies a unique identifier for this table. The name is a community string.
TDomain	Specifies the transport type of the address. Default is snmpUDPDomain.

Variable	Value
TAddress	Specifies the transport address in xx.xx.xx.x x:port format, for example: 10:10:10:10:162, where 162 is the trap listening port on the system 10.10.10.10. You can also specify IPv6 addresses.
Timeout	Specifies the maximum round trip time required for communicating with the transport address. The value is in 1/100 seconds. The default is 1500. When a message is sent to this address and a response (if one is expected) is not received within this time period, an implementation assumes that the response is not delivered.
RetryCount	Specifies the maximum number of retries when a response is not received for a generated message. The count can be in the range of 0–255. The default is 3.
TagList	Contains a list of tag values which are used to select target addresses for a particular operation. A tag refers to a class of targets to which the messages can be sent.
Params	Contains SNMP parameters to be used when generating messages to send to this transport address. For example, to receive SNMPv2C traps use TparamV2.
TMask	Specifies the mask. The value can be empty or in six-byte hex string format. Tmask is an optional parameter that allows an entry in the TargetAddrTable to specify multiple addresses.
MMS	Specifies the maximum message size. The size can be zero, or 484–2147483647. The default is 484. Although the maximum MMS is 2147483647, the switch supports the maximum SNMP packet size of 8192.

Configuring target table parameters

The target table contains the security parameters for SNMP. Configure the target table to set parameters such as SNMP version and security levels by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Edit, SnmpV3**.

2 Double-click **Target Table**.

3 Select the **Target Params Table** tab.

4 Click **Insert**.

5 Type a target table Nname in the **Name** box.

6 From the **MPModel** options, select an SNMP version.

7 From the **Security Model** options, select the security model.

8 In the **SecurityName** box, type **readview** or **writeview**.

9 From the **SecurityLevel** options, select the security level for the table.

10 Click **Insert**.

--End--

Variable definitions

Use the data in the following table to configure a target table with SNMP security parameters.

Variable	Value
Name	Identifies the target table.
MPModel	Specifies the Message Processing Model to use when generating messages: SNMPv1, SNMPv2c, or SNMPv3/USM
SecurityModel	Specifies the security model to use when generating messages: SNMPv1, SNMPv2c, or USM. An implementation can return an inconsistentValue error if an attempt is made to set this variable to a value for a security model which the implementation does not support.
SecurityName	Identifies the Principal on whose behalf SNMP messages are generated.
SecurityLevel	Specifies the security level used when generating SNMP messages: noAuthNoPriv, authNoPriv, or authPriv.

Viewing the trap sender table

Use the Trap Sender Table tab to view source and receiving addresses by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Edit, Chassis**.
2	Click the **Trap Sender Table** tab.

--End--

Variable definitions

Use the data in the following table to use the Trap Sender Table tab.

Variable	Value
RecvAddress	Specifies the IP address for the trap receiver. This is a read-only parameter that contains the IP address configured in the TAddress field in the TargetTable.
SrcAddress	Identifies the IP address for the trap sender.

Configuring an SNMP notify table

Configure the notify table to select management targets to receive notifications, as well as the type of notification to send to each management target.

Configure SNMP notify table by performing this procedure.

Procedure steps

Step	Action
1	In the naviagation tree, open the following folders: **Edit, SnmpV3**.
2	Double-click **Notify Table**.
3	Click **Insert**.
4	Type a notify table name in the **Name** box.
5	Type the transport tag for the table in the **Tag** box.
6	From the **Type** options, select a type.
7	Click **Insert**.

--End--

Variable definitions

Use the data in the following table to configure an SNMP notify table.

Variable	Value
Name	Specifies a unique identifier.
Tag	Specifies the tag.
Type	Determines the type of notification generated. This value is used when generating notifications, and is ignored for other purposes. If an SNMP entity supports only generation of Unconfirmed-Class PDUs then this parameter can be read-only. • trap: messages generated contain Unconfirmed-Class PDUs • inform: messages generated contain Confirmed-Class PDUs.

Configuring SNMP notify filter profile table parameters

Configure the profile table to associate a notification filter profile with a particular set of target parameters by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Edit, SnmpV3**.
2	Double-click **Notify Table**
3	Select the **Notify Filter Profile Table** tab.
4	Click **Insert**.
5	Type a name for the target parameters in the **TargetParamsName** box.
6	Type a name for the notify filter profile in the **NotifyFilterProfileName** box.
7	Click **Insert**.

--End--

Variable definitions

Use the data in the following table to configure a notify filter profile table.

Variable	Value
TargetParamsName	Specifies the unique identifier associated with this entry.
NotifyFilterProfileName	Specifies the name of the filter profile to be used when generating notifications.

Configuring SNMP notify filter table parameters

Configure the SNMP table of filter profiles to determine whether particular management targets receive particular notifications by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Edit, SnmpV3**.
2	Double-click **Notify Table**.
3	Select the **Notify Filter Table** tab.
4	Click **Insert**.
5	In the **NotifyFilterProfileName** box, type a name for the notify filter profile.
6	In the **Subtree** box, type subtree location information in x.x.x.x.x.x.x.x.x. format.
7	In the **Mask** box, type the mask location in hex string format.
8	From the **Type** options, select **included** or **excluded** to set filter flag.
9	Click **Insert**.

--End--

Variable definitions

Use the data in the following table to configure a filter profile.

Variable	Value
NotifyFilterProfileName	Specifies the name of the filter profile used while generating notifications.
Subtree	Specifies the management information base (MIB) subtree which, when combined with Mask, defines a family of subtrees which are included in or excluded from the filter profile. For more information about, see RFC 2573.
Mask	Specifies the bit mask (in hexadecimal) which, in combination with Subtree, defines a family of subtrees which are included in or excluded from the filter profile.
Type	Indicates whether the family of filter subtrees are included in or excluded from a filter.

Enabling SNMP trap logging

You can save a copy of all SNMP traps and view them by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Edit, Diagnostics**.
2	Double-click **General**l.
3	Click the **Error** tab.
4	Select **AuthenticationTraps**.
5	Click **Apply**.
6	To see the Trap log, select **Device**, **Trap Log**.

--End--

Variable definitions

Use the information in the following table to understand error parameters.

Variable	Value
AuthenticationTrap	Enables or disables the sending of traps when an error occurs.
LastErrorCode	Specifies the last reported error code.
LastErrorSeverity	Specifies the last reported error severity. 0= Informative Information 1= Warning Condition 2= Error Condition 3= Manufacturing Information 4= Fatal Condition

Viewing SNMP trap logs

Use logs as part of diagnostic or fault management operations. View SNMP trap logs by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Device, Trap Log**.
2	To export the data to a file, click **Export**.
	--End--

Log configuration

Use log files and messages to help perform diagnostic and fault management functions.

Navigation

- “Configuring the system log” (page 67)
- “Configuring the system log table and severity level mappings” (page 68)
- “Viewing system logs” (page 69)

Configuring the system log

Use the system log to track all user activity on the switch. The system log can send messages to up to ten syslog hosts.
Configure system log by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders:**Edit, Diagnostics System Log**.
2	Double-click **System Log**.
3	Select **Enable**.
4	Configure **MaxHosts** and **Header** as required.
5	Click **Apply**.
	--End--

Variable definitions

Use the information in the following table to help you configure the system log operational parameters.

Variable	Value
Enable	Enables or disables the syslog feature. When enabled, this feature sends a message to a server on a network that is configured to receive and store diagnostic messages from this device. The type of messages sent is user-configurable.
MaxHosts	Specifies the maximum number of remote hosts considered active and able to receive messages from the syslog service.
OperState	Specifies the operational state of the syslog service.
Header	Specifies the IP header type for the syslog packet. The options are: default, managementVIP, and circuitlessIP.

Configuring the system log table and severity level mappings

Use the system log table to customize the mappings between the severity levels and the type of alarms.

Configure system log table and severity level mapping by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Edit, Diagnostics**.
2	Double-click **System Log**.
3	Click the **System Log Table** tab.
4	Click **Insert**.
5	Configure the parameters as required.
6	Click **Insert**.
7	To modify mappings, double-click a parameter to view a list of options. Configure the options as required.
8	Click **Apply**.

--End--

Variable definitions

Use the information in the following table to help you customize severity level mappings.

Variable	Value
Id	Specifies the ID for the syslog host.
IpAddr	Specifies the IP address of the syslog host.
UdpPort	Specifies the UDP port to use to send messages to the syslog host (514–530).
Enable	Enables or disables the sending of messages to the syslog host.
HostFacility	Specifies the syslog host facility used to identify messages (LOCAL0 to LOCAL7). The default is LOCAL7.
Severity	Specifies the message severity for which syslog messages are sent.
MapInfoSeverity	Specifies the syslog severity to use for INFO messages. The default is INFO.
MapWarningSeverity	Specifies the syslog severity to use for WARNING messages. The default is WARNING.
MapErrorSeverity	Specifies the syslog severity to use for ERROR messages. The default is ERROR.
MapFatalSeverity	Specifies the syslog severity to use for FATAL messages. The default is EMERGENCY.

Viewing system logs

Use system logs as part of diagnostic or fault management operations. View system log by performing this procedure.

Procedure steps

Step	Action
1	In the navigation tree, open the following folders: **Device, SysLog**.
2	To export the data to a file, click **Export**.

--End--

Viewing Enterprise Device Manager logs

Enterprise Device Manager logs keep track of all activity using Enterprise Device Manager. Use logs as part of diagnostic or fault management operations.

View Enterprise Device Manager logs by performing this procedure.

Procedure steps

Step	Action
1	In the navigation pane, open the following folders: **Device, Log**.
2	To save the data to a file, click **Save**.
3	To view statistics about the SNMP packets, click **SNMP Stats**.

--End--

Log and trap configuration using the CLI

Use logs and traps to enhance the following:

- fault management operations
- troubleshooting procedures

Navigation

- “SNMP trap configuration” (page 71)
- “Log configuration” (page 85)
- “Configuring CLI logging” (page 96)

SNMP trap configuration

Use Simple Network Management Protocol (SNMP) traps and notifications to allow management stations to gather information about switch activities, alarms, and other information.

In the command line interface (CLI), you configure traps by configuring SNMP trap notifications, creating a target address to which you want to send the notifications, and specifying target parameters.

Specify which protocols and processes generate traps by enabling traps for that protocol. For example, to allow SNMP traps to be generated for Open Shortest Path First (OSPF), use the following command: **`config ip ospf trap enable`**.

For information about configuring SNMP community strings and related topics, see *Nortel Ethernet Routing Switch 8600 Security* (NN46205-601).

Navigation

- "Roadmap of SNMP trap CLI commands" (page 72)
- "Configuring SNMP notifications" (page 74)
- "Configuring an SNMP host target address" (page 76)
- "Configuring SNMP target table parameters" (page 78)
- "Configuring an SNMP notify filter table" (page 80)
- "Configuring SNMP interfaces" (page 81)
- "Enabling SNMP trap logging" (page 82)
- "Configuring a UNIX system log and syslog host" (page 83)

Roadmap of SNMP trap CLI commands

The following roadmap lists some of the CLI commands and their parameters that you can use to complete the procedures in this section.

Command	Parameter
config snmp snmplog	enable <true|false>
	info
	maxfilesize <64 to 256000>
config snmp-v3 notify	create <Notify Name> [tag <value>] [type <value>]
	delete <Notify Name>
	info
	tag <Notify Name> new-tag <value>
	type <Notify Name> new-type <value>
config snmp-v3 ntfy-filter	create <Profile Name> <subtree oid> [mask <value>] [type <value>]
	delete <Profile Name> <subtree oid>
	info
	mask <Profile Name> <subtree oid> new-mask <value>
	type <Profile Name> <subtree oid> new-type <value>
config snmp-v3 ntfy-profile	create <Params Name> [profile <value>]
	delete <Params Name>
	info
	profile <Params Name> <new-profile>

Command	Parameter
config snmp-v3 target-addr	create <Target Name> <Ip addr:port> <Target parm> [timeout <value>] [retry <value>] [taglist <value>] [mask <value>] [mms <value>] [tdomain <value>]
	delete <Target Name>
	info
	mask <Target Name> new-mask <value>
	mms <Target Name> new-mms <value>
	parms <Target Name> new-parms <value>
	retry <Target Name> new-retry <value>
	taglist <Target Name> new-taglist <value>
	timeout <Target Name> new-timeout <value>
config snmp-v3 target-param	create <Tparm Name> mp-model <value> sec-level <value> [sec-name <value>]
	delete <Tparm Name>
	info
	mp-model <Tparm Name> new-mpmodel <value>
	sec-level <Tparm Name> new-seclevel <value>
	sec-name <Tparm Name> [new-secname <value>]
config sys set snmp	agent-conformance <enable|disable>
	force-iphdr-sender <true|false>
	force-trap-sender <true|false>
	info
	sender-ip <ipaddr> <ipaddr>
config sys syslog	info
	ip-header-type <default|circuitless-ip|management-virtual-ip>
	max-hosts <maxhost>
	state <enable|disable>

Command	Parameter
config sys syslog host	address <ipaddr>
	create
	delete
	facility <facility>
	host <enable|disable>
	info
	maperror <level>
	mapfatal <level>
	mapinfo <level>
	mapwarning <level>
	severity <info|warning|error|fatal> [<info|warning|error|fatal>] [<info|warning|error|fatal>] [<info|warning|error|fatal>]
	udp-port <port>
show snmp snmplog	info
show snmplog file	[tail]
	[grep <value>]
show snmp-v3	community
	context
	group-access
	group-member
	mib-view
	notify
	ntfy-filter
	ntfy-profile
	target-addr
	target-param
	usm

Configuring SNMP notifications

Configure the notify table to

- select management targets to receive notifications
- specify the type of notification to send to each management target

Configure SNMP notification by performing this procedure.

Procedure steps

Step	Action
1	Create an SNMP notification: **config snmp-v3 notify create <Notify Name> [tag <value>] [type <value>]**
2	Specify the required tags for an existing notification: **config snmp-v3 notify tag <Notify Name> new-tag <value>**
3	Specify the required type for an existing notification: **config snmp-v3 notify type <Notify Name> new-type <value>**
4	Ensure that the configuration is correct by using one of the following commands: **config snmp-v3 notify info** **show snmp-v3 notify**

--End--

Variable definitions

Use the data in the following table to complete the **config snmp-v3 notify** command.

Variable	Value
create <Notify Name> [tag <value>] [type <value>]	Creates an SNMP trap notification entry. • **<Notify Name>** is the index of the notify table with a string length of 1–32. • **tag <value>** specifies the tag name as a string from 1–255 characters. • **type <value>** specifies the notify type as trap or inform
delete <Notify Name>	Deletes an entry from the notify table. **<Notify Name>** is expressed as a string from 1–32 characters long.
info	Displays the notify table information

Variable	Value
tag <Notify Name> new-tag <value>	Specifies the new notify tag for the entry in the notify table. **new-tag <value>** is expressed as a string from 1–255 characters long.
type <Notify Name> new-type <value>	Specifies the new notify type for the entry in the notify table. **new-type <value>** is expressed as trap or inform.

Configuring an SNMP host target address

Configure a target address to specify the transport addresses to use in the generation of SNMP messages by performing this procedure.

Procedure steps

Step	Action
1	Add an SNMP target address: **ATTENTION** You must include all of the required parameters in this command. If you do not include them, the command is not parsed correctly and the traps are not sent to the destination address. The later addition of these missing parameters does not rectify the situation. **config snmp-v3 target-addr create <Target Name> <Ip addr:port> <Target parm> [timeout <value>] [retry <value>] [taglist <value>] [mask <value>] [mms <value>] [tdomain <value>]**
2	Ensure that the configuration is correct by using one of the following commands: **config snmp-v3 target-addr info** **show snmp-v3 target-addr**

--End--

Variable definitions

Use the data in the following table to use the **config snmp-v3 target-addr** command.

Variable	Value
create <Target Name> <Ip addr:port> <Target parm> [timeout <value>] [retry <value>] [taglist <value>] [mask <value>] [mms <value>] [tdomain <value>]	Creates a new entry for the target address table. • **<Target Name>** is the target name with a string length of 1–32. • **<Ip addr:port>** is the target IP address in the form 1.2.3.4:161 (or ipv6addr:port if the domain option is set to IPv6) with a string length of 1–255. • **<Target parm>** is the target parameter with a string length of 1–32. • **timeout <value>** specifies the timeout value in seconds with a range of 0–2147483647. • **retry <value>** specifies the retry count value with a range of 0–255. • **taglist <value>** specifies the tag list with a string length of 1–255. • **mask <value>** specifies the mask in the form 0x00:00...6 octets separated by colons with a string length of 13–19. • **mms <value>** specifies the maximum message size {0|484–8192} among {0–2147483647} . • **tdomain <value>** specifies the target transport domain.
delete <Target Name>	Deletes an entry from the target address table.
info	Displays target address table information.
mask <Target Name> new-mask <value>	Specifies a new mask for the target.
mms <Target Name> new-mms <value>	Specifies a new maximum message size (MMS) associated with an entry in the target address table. Although the maximum value for the MMS is 2 147 483 647, the device supports the maximum SNMP packet size of 8192 (8K).
parms <Target Name> new-parms <value>	Specifies a new string value that identifies target address table entries.
retry <Target Name> new-retry <value>	Specifies a new number of retries to be attempted when a response is not received for a generated message.

Variable	Value
`taglist <Target Name> new-taglist <value>`	Specifies a new list of tag values.
`timeout <Target Name> new-timeout <value>`	Specifies a new maximum route trip time required for communicating with the transport address.

Example of configuring an SNMP target table

Procedure steps

Step	Action
1	Create the target parameter ID (TparamV2) and target address ID (TAddr1), as well as the other target parameters: `config snmp-v3 target-addr create Taddr1 198.202.188.207:162 TparamV2 timeout 1500 retry 3 taglist DefTag mask ff:ff:00:00:00:00 mms 484`

--End--

Configuring SNMP target table parameters

The target table contains the security parameters for SNMP. Perform this procedure to configure the target table to set parameters such as SNMP version and security levels.

Prerequisites

- To obtain trap configurations in SNMPv1/SNMPv2c/SNMPv3, upgrade to Release 5.0 or greater. Release 3.3 and Release 3.5 support only SNMPv1 or SNMPv2c trap configurations.

Procedure steps

Step	Action
1	Configure SNMP target table parameters by using this command: `config snmp-v3 target-param create <Tparm Name> mp-model <value> sec-level <value> [sec-name <value>]`
2	Ensure that the configuration is correct by using one of the following commands:

```
config snmp-v3 target-param info

show snmp-v3 target-param
```

--End--

Variable definitions

Use the data in the following table to help you use the **config snmp-v3 target-param** command.

Variable	Value
create <Tparm Name> mp-model <value> sec-level <value> [sec-name <value>]	Specifies target table parameters. • **<Tparm Name>** is the name of the target parameter with a string length of 1–32. • **mp-model <value>** specifies the MP model. The valid options are snmpv1, snmpv2c, and usm (SNMPv3). • **sec-level <value>** specifies the security level as noAuthNoPriv, authNoPriv, or authPriv. Optional parameter • **[sec-name <value>** specifies the security name with a string length of 1–32.
delete <Tparm Name>	Deletes the specified target parameter table.
info	Displays information for the target parameter table.
mp-model <Tparm Name> new-mpmodel <value>	Specifies the new SNMP version. The valid options are snmpv1, snmpv2c, and usm (SNMPv3).
sec-level <Tparm Name> new-seclevel <value>	Specifies a new security level. The valid options are noAuthNoPriv, authNoPriv, and authPriv.
sec-name <Tparm Name> [new-secname <value>]	Specifies a new security name (readview or writeview), which identifies the principal that generates SNMP messages.

Example of configuring additional target parameters

Procedure steps

Step	Action
1	Configure target table parameters:

```
config snmp-v3 target-param create TparamV2 mp--model
snmpv2c sec-level noAuthNoPriv sec-name readview
```

--End--

Configuring an SNMP notify filter table

Configure the notify table to select management targets to receive notifications, as well as the type of notification to send to each management target. For more information about the notify filter table, see RFC 3413.

Configure SNMP notify filter table by performing this procedure.

Procedure steps

Step	Action
1	Create a new notify filter table: `config snmp-v3 ntfy-filter create <Profile Name> <subtree oid> [mask <value>] [type <value>]`

2	Ensure that the configuration is correct by using one of the following commands:
	`config snmp-v3 ntfy-filter info`
	`show snmp-v3 ntfy-filter`

--End--

Variable definitions

Use the data in the following table to complete the **`config snmp-v3 ntfy-filter`** command.

Variable	Value
`create <Profile Name> <subtree oid> [mask <value>] [type <value>]`	Creates a notify filter table. • **`<Profile Name>`** specifies the name of the profile with a string length of 1–32. • **`<subtree oid>`** identifies the filter subtree with a string length of 1–32. Optional parameters • **`mask <value>`** specifies the bit mask in combination with snmpNotifyFilterMask, which defines a family of subtrees. Filter mask is

Variable	Value
	expressed as {0x00:00} with a string length from 1–49 characters • **type <value>** indicates whether the family of filter subtrees defined by this entry is included (**include**) or excluded (**exclude**) from a filter.
delete <Profile Name> <subtree oid>	Deletes the specified notify filter profile. • **<Profile Name>** specifies the name of the profile with a string length of 1–32. • **<subtree oid>** identifies the filter subtree with a string length of 1–32.
info	Displays notify filter information.
mask <Profile Name> <subtree oid> new-mask <value>	Specifies the new bit mask in combination with snmpNotifyFilterMask, which defines a family of subtrees. • **<Profile Name>** specifies the name of the profile with a string length of 1–32. • **<subtree oid>** identifies the filter subtree with a string length of 1–32. • **new-mask <value>** is in the format of 0x00:00...with a string length of 1–49.
type <Profile Name> <subtree oid> new-type <value>	Specifies the new type that you want for a profile. The valid values are include and exclude. • **<Profile Name>** specifies the name of the profile with a string length of 1–32. • **<subtree oid>** identifies the filter subtree with a string length of 1–32. • **new-type <value>** specifies include or exclude.

Configuring SNMP interfaces

If the Ethernet Routing Switch 8600 has multiple interfaces, configure the IP interface from which the SNMP traps originate.
Configure SNMP interface by performing this procedure.

Procedure steps

Step	Action
1	Configure the destination and source IP addresses for SNMP traps by using the following commands: **config sys set snmp sender-ip <dest-ipaddr> <src-ipaddr>**

2 If required, send the source address (sender IP) as the sender network in the notification message by using the following commands:

```
config sys set snmp force-trap-sender true
```

3 If required, force the SNMP and IP sender flag to be the same by using the following commands:

```
config sys set snmp force-iphdr-sender true
```

--End--

Variable definitions

Use the information in the following table to complete the **config sys set snmp** command.

Variable	Value
agent-conformance <enable|disable>	Activates or disables the agent conformance mode. Conforms to management information base (MIB) standards when disabled. If you activate this option, feature configuration is stricter and error handling less informative. Activating this option is not a recommended or normally supported mode of operation.
force-iphdr-sender <true|false>	Specify true to configure the SNMP and IP sender to the same value. The default is false.
force-trap-sender <true|false>	Specify true to send the configured source address (sender IP) as the sender network in the notification message.
info	Displays the current SNMP settings.
sender-ip <ipaddr> <ipaddr>	Configures the SNMP trap receiver and source IP addresses. Specify the IP address of the destination SNMP server that receives the SNMP trap notification in the first IP address. Specify the source IP address of the SNMP trap notification packet that is transmitted in the second IP address. If this is set to 0.0.0.0 then the switch uses the IP address of the local interface that is closest (from an IP routing table perspective) to the destination SNMP server.

Enabling SNMP trap logging

Use SNMP trap logging to send a copy of all traps to the Personal Computer Memory Card International Association (PCMCIA) card by performing this procedure.

Prerequisites

- A PCMCIA card must be installed.

Procedure steps

Step	Action
1	Enable SNMP trap logging: **config snmp snmplog enable true**
2	Set the maximum file size: **config snmp snmplog maxfilesize <64-256000>**
3	Ensure that the configuration is correct : **show snmp snmplog info**
4	View the contents of the SNMP log: **show snmplog file [tail] [grep <value>]**
	--End--

Use the information in the following table to help you use the **config snmp snmplog** command.

Table 4
Variable definitions

Variable	Value
enable <true|false>	Enables or disables the logging of traps.
info	Displays information about SNMP logging.
maxfilesize <64-256000>	Specifies the maximum file size for the trap log.

Configuring a UNIX system log and syslog host

The syslog commands control a facility in UNIX machines that logs SNMP messages and assigns each message a severity level based on importance.

Configure UNIX system log and syslog host by performing this procedure.

Procedure steps

Step	Action
1	Configure system logging using the following command, along with the parameters in the following table:

```
config sys syslog
```

2 Configure the syslog host using the following command, along with the parameters in the following table:

```
config sys syslog host
```

3 View the configuration to ensure it is correct by using the following commands:

```
show sys syslog host info
```

```
show sys syslog general-info
```

--End--

Variable definitions

Use the data in the following table to help you use the **config sys syslog** command.

Variable	Value
info	Displays syslog configuration information.
ip-header-type <default|circuitless-ip|management-virtual-ip>	Specifies the IP header in syslog packets to default, circuitless-ip or management-virtual-ip: • If set to default, then for syslog packets that are transmitted in-band through input/output (I/O) ports, the IP address of the VLAN is used. For syslog packets that are transmitted out-of-band through the management port, the physical IP address of the Master CPU is used in the IP header. • If set to management-virtual-ip, then for syslog packets that are transmitted out-of-band only through the management port, the virtual management IP address of the switch is used in the IP header. • If set to circuitless-ip, then for all syslog messages (in-band or out-of-band), the circuitless IP address is used in the IP header. If a user has configured multiple CLIPs, the first CLIP configured is used.
max-hosts <maxhost>	Specifies the maximum number of syslog hosts supported. **<maxhost>** is the maximum number of enabled hosts allowed (1–10).
state <enable|disable>	Enables or disables sending syslog messages on the switch.

Use the data in the following table to help you use the **config sys syslog host** command.

Variable	Value
address <ipaddr>	Configures a host location for the syslog host. **<ipaddr>** is the IP address of the UNIX system syslog host.
create	Creates a syslog host instance.
delete	Deletes a syslog host.
facility <facility>	Specifies the UNIX facility used in messages to the syslog host. **<facility>** is the UNIX system syslog host facility (LOCAL0 to LOCAL7).
host <enable|disable>	Enables or disables the syslog host.
info	Shows information about the syslog host configuration.
maperror <level>	Specifies the syslog severity to use for Error messages. **<level>** is one of {emergency|alert|critical|error|warning|notice|info|debug}.
mapfatal <level>	Specifies the syslog severity to use for Fatal messages. **<level>** is one of {emergency|alert|critical|error|warning|notice|info|debug}.
mapinfo <level>	Specifies the syslog severity level to use for Information messages. **<level>** is one of {emergency|alert|critical|error|warning|notice|info|debug}.
mapwarning <level>	Specifies the syslog severity to use for Warning messages. **<level>** is {emergency|alert|critical|error|warning|notice|info|debug}.
severity <info|warning|error|fatal> [<info|warning|error|fatal>] [<info|warning|error|fatal>] [<info|warning|error|fatal>]	Specifies the severity levels for which syslog messages should be sent for the specified modules. **<severity>** is the severity for which syslog messages are sent.
udp-port <port>	Specifies the UDP port number on which to send syslog messages to the syslog host. **<port>** is the UNIX system syslog host port number (514–530).

Log configuration

Use log files and messages to help perform diagnostic and fault management functions.

Log configuration navigation

- "Roadmap of CLI log commands" (page 86)
- "Configuring logging" (page 87)
- "Viewing logs" (page 88)
- "Configuring the remote host address for log transfer" (page 90)
- "Configuring system logging to a PCMCIA" (page 91)
- "Starting system message logging to a PCMCIA card" (page 93)
- "Starting system message logging to a PCMCIA card" (page 93)
- "Configuring system message control" (page 94)
- "Extending system message control" (page 95)

Roadmap of CLI log commands

The following roadmap lists some of the CLI commands and parameters that you can use to complete the procedures in this section.

Command	Parameter
`config bootconfig logfile <minsize> <maxsize> <maxoccupyPercentage>`	—
`config bootconfig flags logging <true|false>`	—
`config log`	`clear`
	`info`
	`level [<level>]`
	`logToPCMCIA <true|false>`
	`screen [<setting>]`
	`write <str>`
`config log transferFile`	`add-IP <ipaddr>`
	`filename <str>`
	`info`
	`remove-IP`

Command	Parameter
config sys set msg-control	action <suppress-msg|send-trap|both>
	control-interval <minutes>
	disable
	enable
	info
	max-msg-num <number>
config sys set msg-control force-msg	add <str>
	del <str>
	info
show log file [tail] [name-of-file <value>] [category <value>] [severity <value>] [CPU <value>] [save-to-file <value>]	**ATTENTION** The **show log file tail name-of-file <filename>** command does not produce any output if the **tail** option is used. The workaround is to redirect the output to another file using the save-to-file option and view the log file in a text editor.
show log level	—

Configuring logging

You can configure log file parameters, as well as write, or clear the log file automatically created by the system by performing this procedure.

Procedure steps

Step	Action
1	Define which messages are logged: **config log level [<level>]**
2	Write the log file from memory to a file: **config log write <str>**
3	Use the following table to help you configure other parameters as required.

--End--

Variable definitions

Use the information in the following table to help you use the **config log** commands.

Variable	Value
clear	Clears the log file.
info	Displays the current log settings.
level [<level>]	Shows and sets the logging level. **<level>** is one of these values: 0 = Information; all messages are recorded. 1 = Warning; records only warning and more serious messages. 2 = Error; records only error and more serious messages. 3 = Manufacturing; this parameter is not available for customer use. 4 = Fatal; records only fatal messages.
logToPCMCIA <true|false>	Starts or stops logging system messages to the PCMCIA card.
screen [<setting>]	Sets the log display on the screen to on or off, where setting is on or off.
write <str>	Writes the log file with the designated string. **<str>** is the string or command that you append to the log file. If the string contains spaces, you must enclose the string in quotation marks.

Viewing logs

Log files can be viewed by file name, category, severity, and SF/CPU by performing this procedure.

ATTENTION
The **show log file tail name-of-file <filename>** command does not produce any output if the **tail** option is used. The workaround is to redirect the output to another file using the save-to-file option and view the log file in a text editor.

Procedure steps

Step	Action
1	Display log information by file name, category, severity, or SF/CPU: **show log file [tail] [name-of-file <value>] [category <value>] [severity <value>] [CPU <value>] [save-to-file <value>]**

--End--

Variable definitions

Use the following table for help with the **show log file** command.

Variable	Value
category <value>	Filters and list the logs according to category. Specify a string length of 0–100 characters. To specify multiple filters, separate each category by the vertical bar (**|**), for example, **OSPF | FILTER | QOS**. Options include ATM, CPU, DVMRP, EAP, FILTER, HW, IGMP, IP, IPX, IP-RIP, IPMC, MLT, MPLS, OSPF , PIM, POLICY, POS, QOS, RADIUS, RIP, RMON, SNMP, STG, SW, VLAN, WEB, COP-SW, HAL, RCMPLS.
CPU <value>	Filters and list the logs according to the SF/CPU that generated it. Specify a string length of 0..25 characters. To specify multiple filters, separate each SF/CPU by the vertical bar (**|**), for example, **CPU3 | CPU5 | CPU6**.
name-of-file <value>	Displays the valid logs from the file name specified by <value>. For example, /pcmcia/logcopy.txt. You cannot use this command on the current log file—the file into which the messages are currently logged. Specify a string length of 1–99 characters.
save-to-file <value>	Redirects the output to the specified file and remove all encrypted information. The tail option is not supported with the **save-to-file** option. Specify a string length of 0–99 characters.
severity <value>	Filters and list the logs according to severity. Express the value as a string from 0–25 characters long. Specify INFO, ERROR, WARNING, or FATAL. To specify multiple filters, separate each severity by the vertical bar (**|**), for example, **ERROR | WARNING | FATAL**.
tail	Displays file from tail.

Job aid: show log file example

The following example shows you how to display all of the log messages generated by OSPF and IP with severity levels of ERROR and WARNING.

```
ERS 8610:5# show log file category OSPF | IP severity
ERROR | WARNING cpu CPU5
```

The following example shows you how to display the log messages from a specific log file.

```
ERS 8610:5# show log file name-of-file /pcmcia/sample.txt
```

Configuring the remote host address for log transfer

Configure the remote host address for log transfer. The system transfers the current log file to a remote host when the log file size reaches the configured maximum size.

Configure remote host address for log transfer by performing this procedure.

Prerequisites

- The IP address you configure for the remote host must be reachable at the time of configuration.

Procedure steps

Step	Action
1	Configure the remote host address for log transfer: **config log transferFile <id> add_IP <ipaddr>** **<id>** specifies the ID for the remote host. The range is 1–10.
2	You can specify the file name: **config log transferFile <id> filename <str>** This command sets the IP address for the remote host to the default (0.0.0.0).
3	Show the configured IP address and the file name for the remote host: **config log transferFile <id> info**

--End--

Variable definitions

Use the information in the following table to help you use the **config log transferFile <id>** command.

Variable	Value
add-IP <ipaddr>	Specifies the IP address of the host to where the log file needs to be transferred. Specify the IP address in the format a.b.c.d. The remote host must be reachable or the configuration fails.

Variable	Value
filename <str>	Specify the name of the file stored in the remote host. If not configured, the current log file name is the default. **ATTENTION** Nortel recommends that you do not set this option. If this option is set, the previously transferred log file is overwritten on the remote server.
info	Displays information about the log file transfer configuration.
remove-IP	Removes the IP address.

Job aid: example of config log transferFile command

The following example shows you how to configure the remote host address for log transfer.

```
ERS-8610:5# config log transferFile 1 add-IP 10.10.42.1

ERS-8610:5# config log transferFile 1 info

Sub-Context:
Current Context:
RemoteIPAddress :  10.10.42.1
File Name :  39d00005.000
```

If the IP address you are attempting to configure is not reachable, the following message is displayed:

```
Destination IP address not reachable !!!  Could not configure
```

Configuring system logging to a PCMCIA

System logs are a valuable diagnostic tool. You can send log messages to a PCMCIA card for later retrieval.

Define the minimum and maximum log file sizes to bound the file storage size on the PCMCIA card. The system transfers the current log file to a remote host when the log file size reaches the configured maximum size.

Although log file parameters are stored in the boot configuration file, you can change them at anytime without rebooting the system. Changes made to these parameters take effect immediately.

When you remove the PCMCIA card from the primary SF/CPU, a trap is generated and system logging continues only in DRAM .

CAUTION
Risk of data loss
Before removing the PCMCIA card from your primary SF/CPU, you must stop the logging of system messages. Failure to do so can corrupt the file system on the PCMCIA card and cause your log file to be permanently lost.

Prerequisites

- A PCMCIA card must be installed.

Procedure steps

Step	Action
1	Enable system logging to a PCMCIA card: **`config bootconfig flags logging <true|false>`** If the logging flag is not set to true, the entries are stored in memory.
2	Configure the logfile parameters: **`config bootconfig logfile <minsize> <maxsize> <maxoccupyPercentage>`**

--End--

Variable definitions

Use the data in the following table to help you use the **`config bootconfig`** commands in this procedure.

Variable	Value
`flags logging <true|false>`	Enables or disables logging to a PCMCIA card. The log file is named using an 8.3 (xxxxxxxx.sss) format. The first six characters of the file name contain the last three bytes of the chassis base MAC address. The next two characters specify the slot number of the SF/CPU that generated the logs. The last three characters denote the sequence number of the log file. Multiple sequence numbers are generated for the same chassis and same

Variable	Value
	slot, if the SF/CPU is replaced, reinserted, or if the maximum log file size is reached.
logfile <minsize> <maxsize> <maxoccupyPercentage>	Configures the logfile parameters: • **<minsize>** specifies the minimum space used for the logfile from 64–500 KB. • **<maxsize>** specifies the minimum space used for the logfile from 500–16384 KB. • **<maxoccupyPercentage>** specifies the maximum percentage of PCMCIA space used for the logfile from 10–90%.

Starting system message logging to a PCMCIA card

Begin or stop logging system messages to the PCMCIA card.

When you remove the PCMCIA card from the primary SF/CPU, a trap is generated and system logging continues only in DRAM.

CAUTION
Risk of data loss
Before removing the PCMCIA card from your primary SF/CPU, you must stop the logging of system messages. Failure to do so can corrupt the file system on the PCMCIA card and cause your log file to be permanently lost.

Procedure steps

Step	Action
1	Start or stop logging system messages on the PCMCIA card: **config log logToPCMCIA <true|false>**

--End--

Variable definitions

Use the date in the following table to complete the **config log logToPCMCIA** command.

Variable	Value
<true|false>	Starts or stop the logging of system messages on the PCMCIA card. If true is specified, the following message appears: `Logging to PCMCIA STARTED`. If false is specified, the following message appears: `Logging to PCMCIA STOPPED`.

Configuring system message control

Configure system message control to suppress duplicate error messages on the console, and to determine the action to take if they occur. Configure system message control by performing this procedure.

Procedure steps

Step	Action
1	Configure system message control action: **config sys set msg-control action <suppress-msg|send-trap|both>**
2	Configure the maximum number of messages: **config sys set msg-control max-msg-num <number>**
3	Configure the interval: **config sys set msg-control control-interval <minutes>**
4	Enable message control: **config sys set msg-control enable**

--End--

Variable definitions

Use the information in the following table to complete the **config sys set msg-control** command.

Variable	Value
action <suppress-msg|send-trap|both>	Configures the message control action.
control-interval <minutes>	Configures the message control interval in minutes. The valid options are 1–30.
disable	Disables system message control.
enable	Activates system message control. Enabling this command suppresses duplicate error messages.

Variable	Value
info	Displays the configuration of system message control.
max-msg-num <number>	Configures the number of occurrences of a message after which the control action happens. To set the maximum number of occurrences, enter a value from 2–500.

Extending system message control

Use the force message control option to extend the message control feature functionality to the software and hardware log messages.

To enable the message control feature, you must specify an action, control interval, and maximum message number. After enabling the feature, the log messages, which get repeated and cross the maximum message number in the control interval, trigger the force message feature. You can either suppress the message or send a trap notification, or both. Extend system message control by performing this procedure.

Procedure steps

Step	Action
1	Configure the force message control option: **config sys set msg-control force-msg add <str>**
2	Ensure the configuration is correct: **config sys set msg-control force-msg info**

--End--

Variable definitions

Use the information in the following table to complete the **config sys set msg-control force-msg** command.

Variable	Value
add <str>	Used to add a forced message control pattern, where **<str>** is a string of 4 characters. You can add a four-byte pattern into the force-msg table. The software and the hardware log messages that use the first four bytes matching one of the patterns in the force-msg table undergo the configured message control action. You can specify up to 32 different patterns in the force-msg

Variable	Value
	table. This includes a wild-card pattern (****) as well. If you specify the wild-card pattern, all messages undergo message control.
del <str>	Deletes a forced message control pattern.
info	Displays the current configuration.

Configuring CLI logging

When enabled, CLI logging keeps track of all command line interface commands executed on the switch. Use CLI logging for fault management purposes.

Configure CLI logging by performing this procedure.

Procedure steps

Step	Action
1	Enable or disable CLI logging: **config cli clilog enable <true|false>**
2	Change the maximum file size used for CLI logs: **config cli clilog maxfilesize <64 to 256000>**
3	Ensure that the configuration is correct: **config cli clilog info** **show cli clilog info**
4	View the CLI log: **show clilog file [tail] [grep <value>]**

--End--

Variable definitions

Use the information in the following table to help you use the **config cli clilog** commands.

Variable	Value
enable <true|false>	Enables or disables CLI logging.
info	Shows configuration information.
maxfilesize <64 to 256000>	Specifies the maximum file size of the log file in KB.

Use the information in the following table to help you use the **show clilog file** commands.

Variable	Value
tail	Shows the last results first.
grep <value>	Performs a string search in the CLI log file. **<value>** is the string, of up to 256 characters in length, to match.

Log and trap configuration using the NNCLI

Use logs and traps as part of fault management operations and to provide diagnostic information in troubleshooting procedures.

Navigation

- "SNMP trap configuration" (page 99)
- "Log configuration" (page 111)
- "Configuring NNCLI logging" (page 120)

SNMP trap configuration

Use Simple Network Management Protocol (SNMP) traps and notifications to allow management stations to gather information about switch activities, alarms, and other information.

Specify which protocols and processes generate traps by enabling traps for that protocol. For example, to allow SNMP traps to be generated for Open Shortest Path First (OSPF), use the following command: **`ip ospf trap enable`**.

For information about configuring SNMP community strings and related topics, see *Nortel Ethernet Routing Switch 8600 Security* (NN46205-601).

Navigation

- "Roadmap of SNMP trap NNCLI commands" (page 100)
- "Job aid: SNMP configuration in the NNCLI" (page 101)
- "Configuring SNMP notifications" (page 103)
- "Configuring an SNMP host" (page 103)
- "Configuring SNMP target table parameters" (page 106)
- "Configuring an SNMP notify filter table" (page 106)

- “Configuring SNMP interfaces” (page 106)
- “Enabling SNMP trap logging” (page 108)
- “Configuring a UNIX system log and syslog host” (page 109)

Roadmap of SNMP trap NNCLI commands

The following roadmap lists some of the Nortel Networks command line interface (NNCLI) commands and parameters that you can use to complete the procedures in this section.

Command	Parameter
Privileged EXEC mode	
`clear logging`	—
`show snmp-server host`	—
`show snmp-server notify-filter`	—
`show syslog`	—
`show syslog host <1-10>`	—
Global Configuration mode	
`snmp-server`	`agent-conformance enable`
	`authentication-trap enable`
	`bootstrap`
	`community`
	`contact`
	`force-iphdr-sender enable`
	`force-trap-sender enable`
	`group`
	`host`
	`location`
	`name`
	`notify-filter <WORD 1-32> <WORD 1-32>`
	`sender-ip <A.B.C.D> <A.B.C.D>`
	`user`
	`view`

Command	Parameter
snmp-server host <WORD 1-256> port <1-65535>	v1 <WORD 1-32> [filter <WORD 1-32>] [target-name <WORD 1-32>]
	v2c <WORD 1-32> [inform [mms <0-2147 483647>] [retries <0-255>] [timeout <0-2147483647>]] [filter <WORD 1-32>] [target-name <WORD 1-32>]
	v3 {noAuthnoPriv|authNoPriv|authPriv} <WORD 1-32> [inform [retries <0-255>] [timeout <0-2147483647>]] [filter <WORD 1-32>] [target-name <WORD 1-32>]
syslog	enable
	host
	ip-header-type <default|circuitless-ip|management-virtual-ip>
	max-hosts <1-10>
syslog host <1-10>	address <A.B.C.D>
	enable
	facility {local0|local1|local2|local3|local4|local5|local6|local7}
	maperror {emergency|alert|critical|error|warning|notice|info|debug}
	mapfatal {emergency|alert|critical|error|warning|notice|info|debug}
	mapinfo {emergency|alert|critical|error|warning|notice|info|debug}
	mapwarning {emergency|alert|critical|error|warning|notice|info|debug}
	severity <info|warning|error|fatal> [<info|warning|error|fatal>] [<info|warning|error|fatal>] [<info|warning|error|fatal>]
	udp-port <514-530>

Job aid: SNMP configuration in the NNCLI

SNMP is configured differently in the Nortel Networks command line interface (NNCLI) than in the command line interface (CLI). Auto-generation of several parameters and command structure changes means that several configuration procedures are no longer required in the NNCLI. The following sections describe the changes.

- “snmpNotifyFilterTable” (page 102)
- “snmpTargetAddrTable” (page 102)

- "snmpTargetParamsTable" (page 103)
- "snmpNotifyTable" (page 103)

snmpNotifyFilterTable

In the CLI, the Type is explicitly specified to be include or exclude. In the NNCLI, this is specified by using the Subtree object identifier (OID). If the Subtree OID parameter uses a '+' prefix (or no prefix), this indicates include. If the Subtree OID uses the '-' prefix, this indicates exclude.

In the CLI, the Mask is explicitly configured in hex-colon format. In NNCLI, the user does not calculate the mask, because it is automatically calculated. The wildcard character '*' can be used to specify the mask within the OID. The OID need not be specified in the dotted decimal format; you can alternatively specify the management information base (MIB) parameter names. The OIDs are automatically calculated.

Example:

```
snmp-server view abc ifEntry.*.2
```

This command creates an entry with ViewName = abc, Subtree = 1.3.6.1.2.1.2.2.1.0.2 and Mask = FF: A0.

Notify-filter mask entries in the notify-filter table are not saved if you change from CLI to NNCLI mode.

snmpTargetAddrTable

In the CLI, the TargetName is user-configurable. In NNCLI, it is generated based on the TargetAddress, SecurityModel and SecurityName given by the user while creating an entry.

The TargetAddrTaglist can be specified only for v2 and v3 users. If the Inform parameter is not configured, the default is used (Trap).

In NNCLI, it is not possible to modify the timeout, retries and MMS values for an SNMPv1 target-address, but is possible for SNMPv2 and SNMPv3. The port option is not required for snmp-server host creation.

In the NNCLI, the TargetAddrParamsName is the same as the TargetName. In CLI, the user specifies both of these names explicitly. The original TargetName is retained across CLI and NNCLI.

For successful load of SNMP server host configurations into NNCLI from CLI or Device Manager, those configurations must be complete. That is, the corresponding TargetParam entries and TargetAddress configurations must be complete.

In the NNCLI, the snmpTargetAddrTable, snmpNotifyFilterProfileTable, and snmpTargetParamsTable are simultaneously created using the **snmp-server host** command. Deletion of an entry in the snmpTargetAddrTable deletes all the entries corresponding to that entry from these tables.

Toggling between CLI and NNCLI can cause loss of configurations because the target address table configurations are different in CLI and NNCLI. The Tparm Name parameter is lost while changing from CLI to NNCLI.

snmpTargetParamsTable

In the NNCLI, the snmpTargetParamsTable is populated by using the **snmp-server host** command.

snmpNotifyTable

There are two preconfigured entries in the snmpNotifyTable. These entries cannot be modified or deleted. The NNCLI command set does not allow you to create or delete entries in the snmpNotifyTable; this table is automatically generated.

Configuring SNMP notifications

The SNMP notification table (snmpNotifyTable) is preconfigured and nonconfigurable in the NNCLI.

Configuring an SNMP host

Configure an SNMP host so that the switch can forward SNMP traps to a host for monitoring by performing this procedure. You can use SNMPv1, SNMPv2c, or SNMPv3.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Configure an SNMPv1 host: **snmp-server host <WORD 1-256> port <1-65535> v1 <WORD 1-32> [filter <WORD 1-32>] [target-name <WORD 1-32>]** **<WORD 1-256>** specifies either an IPv4 or IPv6 address. **port <1-65535>** specifies the host server port number.
2	Configure an SNMPv2c host:

```
snmp-server host <WORD 1-256> port <1-65535> v2c
<WORD 1-32> [inform [mms <0-2147483647>] [retries
<0-255>] [timeout <0-2147483647>]] [filter <WORD 1-32>]
[target-name <WORD 1-32>]
```

3 Configure an SNMPv3 host:

```
snmp-server host <WORD 1-256> port <1-65535> v3
{noAuthnoPriv|authNoPriv|AuthPriv} <WORD 1-32> [inform
[retries <0-255>] [timeout <0-2147483647>]] [filter
<WORD 1-32>] [target-name <WORD 1-32>]
```

4 Ensure that the configuration is correct:

```
show snmp-server host
```

--End--

Variable definitions

Use the data in the following table to use the **snmp-server host <WORD 1-256> port <1-65535>** command.

Variable	Value
v1 <WORD 1-32> [filter <WORD 1-32>] [target-name <WORD 1-32>]	Creates a new SNMPv1 entry for the target address table. • **<WORD 1-32>** specifies the security name, which identifies the principal that generates SNMP messages. • **filter <WORD 1-32>** specifies the filter profile to use. • **target-name <WORD 1-32>** is the target name with a string length of 1–32.
v2c <WORD 1-32> [inform [mms <0-2147483647>] [retries <0-255>] [timeout <0-2147483647>]] [filter <WORD 1-32>] [target-name <WORD 1-32>]	Creates a new SNMPv2c entry for the target address table. • **<WORD 1-32>** specifies the security name, which identifies the principal that generates SNMP messages. • **inform** indicates that SNMP notifications should be sent as inform (rather than trap). • **mms <0-2147483647>** specifies the maximum message size as an integer with a range of 1–2147483647. • **retries <0-255>** specifies the retry count value with a range of 0–255. • **timeout <0-2147483647>** specifies the timeout value in seconds with a range of 0–214748364.

Variable	Value
	• **filter <WORD 1-32>** specifies the filter profile to use. • **target-name <WORD 1-32>** is the target name with a string length of 1–32.
v3 {noAuthnoPriv|authNoPriv|AuthPriv} <WORD 1-32> [inform [retries <0-255>] [timeout <0-2147483647>]] [filter <WORD 1-32>] [target-name <WORD 1-32>]	Creates a new SNMPv3 entry for the target address table. • **{noAuthnoPriv|authNoPriv|AuthPriv}** specifies the security level. • **<WORD 1-32>** specifies the security name, which identifies the principal that generates SNMP messages. • **inform** indicates that SNMP notifications should be sent as inform (rather than trap). • **retries <0-255>** specifies the retry count value with a range of 0–255. • **timeout <0-2147483647>** specifies the timeout value in seconds with a range of 0–214748364. • **filter <WORD 1-32>** specifies the filter profile to use. • **target-name <WORD 1-32>** is the target name with a string length of 1–32.

Example of configuring an SNMP host

Procedure steps

Step	Action
1	Configure the target table entry: **snmp-server host 198.202.188.207 port 162 v2c ReadView inform retries 3** **snmp-server host 198.202.188.207 port 162 v2c ReadView inform mms 484** **snmp-server host 198.202.188.207 port 162 v2c ReadView inform timeout 1500**

--End--

Configuring SNMP target table parameters

In NNCLI, the target table parameters (security name, model) are configured as part of the SNMP host configuration. For more information about, see “Configuring an SNMP host” (page 103).

Configuring an SNMP notify filter table

Configure the notify table to select management targets to receive notifications, as well as the type of notification to send to each management target. For more information about the notify filter table, see RFC 3413.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Create a new notify filter table: **snmp-server notify-filter <WORD 1-32> <WORD 1-32>**
2	Ensure that the configuration is correct: **show snmp-server notify-filter**

--End--

Variable definitions

Use the data in the following table to complete the **snmp-server notify-filter** command.

Variable	Value
<WORD 1-32> <WORD 1-32>	Creates a notify filter table. • **<WORD 1-32>** specifies the name of the filter profile with a string length of 1–32. • The second **<WORD 1-32>** identifies the filter subtree OID with a string length of 1–32. If the Subtree OID parameter uses a ’+’ prefix (or no prefix), this indicates include. If the Subtree OID uses the ‘-‘ prefix, this indicates exclude.

Configuring SNMP interfaces

If the Ethernet Routing Switch 8600 has multiple interfaces, configure the IP interface from which the SNMP traps originate.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Configure the destination and source IP addresses for SNMP traps: **snmp-server sender-ip <A.B.C.D> <A.B.C.D>**
2	If required, send the source address (sender IP) as the sender network in the notification message: **snmp-server force-trap-sender enable**
3	If required, force the SNMP and IP sender flag to be the same: **snmp-server force-iphdr-sender enable**
	--End--

Variable definitions

Use the information in the following table to complete the **snmp-server** command.

Variable	Value
agent-conformance enable	Enables the agent conformance mode. Conforms to management information base (MIB) standards when disabled. If you activate this option, feature configuration is stricter and error handling less informative. Activating this option is not a recommended or normally supported mode of operation.
authentication-trap enable	Activates the generation of authentication traps.
bootstrap	Sets SNMP initial user entry.
community	Sets community table.
contact	Specifies the text for the MIB object sysContact.
force-iphdr-sender enable	Enables the automatic configuration of the SNMP and IP sender to the same value. The default is false.

Variable	Value
force-trap-sender enable	Enabled sending the configured source address (sender IP) as the sender network in the notification message.
group	Sets the SNMP v3 group access table.
host	Specifies hosts to receive SNMP notifications.
location	Specifies the text for the MIB object sysLocation.
name	Specifies the text for the MIB object sysName.
notify-filter	Creates a new entry for the notify filter table.
sender-ip <A.B.C.D> <A.B.C.D>	Configures the SNMP trap receiver and source IP addresses. Specify the IP address of the destination SNMP server that receives the SNMP trap notification in the first IP address. Specify the source IP address of the SNMP trap notification packet that is transmitted in the second IP address. If this is set to 0.0.0.0 then the switch uses the IP address of the local interface that is closest (from an IP routing table perspective) to the destination SNMP server.
user	Create or modify an SNMPv3 user.
view	Create or modify an SNMP access view.

Enabling SNMP trap logging

Use SNMP trap logging to send a copy of all traps to the Personal Computer Memory Card International Association (PCMCIA) card. Enable SNMP trap logging by performing this procedure.

Prerequisites

- Access Global Configuration mode.
- A PCMCIA card must be installed.

Procedure steps

Step	Action
1	Enable SNMP trap logging: **snmplog enable**
2	Configure the maximum log file size: **snmplog maxfilesize <64-256000>**

3 View the contents of the SNMP log:

```
show snmplog
```

--End--

Variable definitions

Use the information in the following table to help you use the **snmplog** command.

Variable	Value
enable	Enables or disables the logging of traps.
maxfilesize <64-256000>	Specifies the maximum file size, in kilobytes, for the trap log.

Configuring a UNIX system log and syslog host

The syslog commands control a facility in UNIX machines that logs SNMP messages and assigns each message a severity level based on importance.

Configure UNIX sytem log and syslog host by performing this procedure.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Enable the system log: **syslog enable** Configure other syslog parameters as required using the parameters in the following table.
2	Configure the syslog host: **syslog host <1-10>** Configure other syslog host parameters as required using the parameters in the following table.
3	View the configuration to ensure it is correct: **show syslog**

```
show syslog host <1-10>
```

--End--

Variable definitions

Use the data in the following table to help you use the **syslog** command.

Variable	Value
enable	Enables the sending of syslog messages on the switch.
host	Specifies the settings for each host.
ip-header-type <default|circuitless-ip|management-virtual-ip>	Specifies the IP header in syslog packets to default, circuitless-ip or management-virtual-ip. • If set to default, then for syslog packets that are transmitted in-band through input/output (I/O) ports, the IP address of the VLAN is used. For syslog packets that are transmitted out-of-band through the management port, the physical IP address of the Master CPU is used in the IP header. • If set to management-virtual-ip, then for syslog packets that are transmitted out-of-band only through the management port, the virtual management IP address of the switch is used in the IP header. • If set to circuitless-ip, then for all syslog messages (in-band or out-of-band), the circuitless IP address is used in the IP header. If a user has configured multiple CLIPs, the first CLIP configured is used.
max-hosts <1-10>	Specifies the maximum number of syslog hosts supported. **<maxhost>** is the maximum number of enabled hosts allowed (1– 10).

Use the data in the following table to help you use the **syslog host <1-10>** command.

Variable	Value
address <A.B.C.D>	Configures a host location for the syslog host. **<A.B.C.D>** is the IP address of the UNIX system syslog host.

Variable	Value
facility {local0|local1|local2|local3|local4|local5|local6|local7}	Specifies the UNIX facility used in messages to the syslog host. **{local0|local1|local2|local3|local4|local5|local6|local7}** is the UNIX system syslog host facility (LOCAL0 to LOCAL7).
enable	Enables the syslog host.
maperror {emergency|alert|critical|error|warning|notice|info|debug}	Specifies the syslog severity to use for Error messages.
mapfatal {emergency|alert|critical|error|warning|notice|info|debug}	Specifies the syslog severity to use for Fatal messages.
mapinfo {emergency|alert|critical|error|warning|notice|info|debug}	Specifies the syslog severity level to use for Information messages.
mapwarning {emergency|alert|critical|error|warning|notice|info|debug}	Specifies the syslog severity to use for Warning messages.
severity <info|warning|error|fatal> [<info|warning|error|fatal>] [<info|warning|error|fatal>] [<info|warning|error|fatal>]	Specifies the severity levels for which syslog messages should be sent for the specified modules.
udp-port <514-530>	Specifies the UDP port number on which to send syslog messages to the syslog host. This is the UNIX system syslog host port number (514–530).

Log configuration

Use log files and messages to help perform diagnostic and fault management functions.

Log configuration navigation

- “Roadmap of NNCLI log commands” (page 112)
- “Configuring logging” (page 112)
- “Viewing logs” (page 114)
- “Configuring the remote host address for log transfer” (page 115)
- “Configuring system logging to a PCMCIA” (page 116)

- “Starting system message logging to a PCMCIA card” (page 118)
- “Configuring system message control” (page 119)
- “Extending system message control” (page 119)

Roadmap of NNCLI log commands

The following roadmap lists some of the NNCLI commands and their parameters that you can use to complete the procedures in this section.

Command	Parameter
Privileged EXEC mode	
`clear logging`	—
`show logging`	`config`
	`file [tail] [category <WORD 0-100>] [severity <WORD 0-25>] [CPU <WORD 0-25>] [name-of-file <WORD 1-99>] [save-to-file <WORD 1-99>]`
	`level`
	`transferFile <1-10>`
Global Configuration mode	
`boot config logfile <64-500> <500-16384> <10-90>`	—
`boot config flags logging`	—
`logging`	`level <0-4>`
	`logToPCMCIA`
	`screen`
	`TransferFile <1-10>`
	`write <WORD 1-1536>`
`logging transferFile <1-10>`	`address <A.B.C.D>`
	`filename <WORD 0-255>`
`sys msg-control`	`action <suppress-msg|send-trap|both>`
	`control-interval <1-30>`
	`max-msg-num <2-500>`
`sys force-msg <WORD 4-4>`	—

Configuring logging

You can configure log file parameters, as well as write, or clear the log file automatically created by the system by performign this procedure.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Define which messages are logged: **logging level <0-4>**
2	Write the log file from memory to a file: **logging write <WORD 1-1536>**
3	Use the following table to help you configure other parameters as required.
	--End--

Variable definitions

Use the information in the following table to help you use the **logging** commands.

Variable	Value
level <0-4>	Shows and sets the logging level. The level is one of these values: 0 = Information; all messages are recorded. 1 = Warning; only warning and more serious messages are recorded. 2 = Error; only error and more serious messages are recorded. 3 = Manufacturing; this parameter is not available for customer use. 4 = Fatal; only fatal messages are recorded.
logToPCMCIA	Starts logging system messages to the PCMCIA card.
screen	Sets the log display on the screen to on.

Variable	Value
transferFile <1-10> <address> <WORD 0-255>	Specifies the file ID expressed as an integer from 1–10. • **address** is the IP address expressed as {A.B.C.D} • **WORD <0-255>** is the file name expressed as an integer from 0–255
write <WORD 1-1536>	Writes the log file with the designated string. **<WORD 1-1536>** is the string or command that you append to the log file. If the string contains spaces, you must enclose the string in quotation marks.

Viewing logs

Log files can be viewed by file name, category, severity, and SF/CPU. View log files by performing this procedure.

Prerequisites

- Access Privileged EXEC mode.

Procedure steps

Step	Action
1	Display log information by file name, category, severity, or SF/CPU: **show logging file [tail] [category <WORD 0-100>] [severity <WORD 0-25>] [CPU <WORD 0-25>] [name-of-file <WORD 1-99>] [save-to-file <WORD 1-99>]**

--End--

Variable definitions

Use the following table for help with the **show logging file** command.

category <WORD 0-100>	Filters and list the logs according to category. Specify a string length of 0–100 characters. Categories include SNMP, EAP, RADIUS, RMON, WEB, STG, IGMP, HW, MLT, FILTER, QOS, SW, CPU, IP, VLAN, IPMC, ATM, DVMRP, |IPX, IP-RIP, MPLS, OSPF, |PIM, POLICY, POS, RIP. To specify multiple filters, separate each category by the vertical bar (**|**), for example, **OSPF | FILTER | QOS**.
CPU <WORD 0-25>	Filters and list the logs according to the SF/CPU that generated it. Specify a string length of 0–25 characters. To specify multiple filters, separate each SF/CPU by the vertical bar (**|**), for example, **CPU5 | CPU6**.
name-of-file <WORD 1-99>	Displays the valid logs from this file. For example, /pcmcia/logcopy.txt. You cannot use this command on the current log file—the file into which the messages are currently logged). Specify a string length of 1–99 characters.
save-to-file <WORD 1-99>	Redirects the output to the specified file and remove all encrypted information. The tail option is not supported with the **save-to-file** option. Specify a string length of 1–99 characters.
severity <WORD 0-25>	Filters and list the logs according to severity. Choices include INFO, ERROR, WARNING, FATAL. To specify multiple filters, separate each severity by the vertical bar (**|**), for example, **ERROR | WARNING | FATAL**.

Configuring the remote host address for log transfer

Configure the remote host address for log transfer. The system transfers the current log file to a remote host when the log file size reaches the configured maximum size.

Configure the remote host address for log transfer by performing this procedure.

Prerequisites

- The IP address you configure for the remote host must be reachable at the time of configuration.
- Access Global Configuration mode.

Procedure steps

Step	Action
1	Configure the remote host address for log transfer:

```
logging transferFile <1-10> address <A.B.C.D>
```

2 Specify the file name:

```
logging transferFile <1-10> filename <WORD 0-255>
```

--End--

Variable definitions

Use the information in the following table to help you use the **`logging transferFile <1-10>`** command.

Variable	Value
`address <A.B.C.D>`	Specifies the IP address of the host to where the log file needs to be transferred. The remote host must be reachable or the configuration fails.
`filename <WORD 0-255>`	Specify the name of the file stored in the remote host. If not configured, the current log file name is the default. **ATTENTION** Nortel recommends that you do not set this option. If this option is set, the previously transferred log file is overwritten on the remote server.

Configuring system logging to a PCMCIA

System logs are a valuable diagnostic tool. You can send log messages to a PCMCIA card for later retrieval.

Define the minimum and maximum log file sizes to bound the file storage size on the PCMCIA card. The system transfers the current log file to a remote host when the log file size reaches the configured maximum size.

Although log file parameters are stored in the boot configuration file, you can change them at anytime without rebooting the system. Changes made to these parameters take effect immediately.

When you remove the PCMCIA card from the primary SF/CPU, a trap is generated and system logging continues only in DRAM.
Configure system logging to a PCMCIA by performing this procedure.

CAUTION
Risk of data loss
Before removing the PCMCIA card from your primary SF/CPU, you must stop the logging of system messages. Failure to do so can corrupt the file system on the PCMCIA card and cause your log file to be permanently lost.

Prerequisites

- A PCMCIA card must be installed.
- Access Global Configuration mode.

Procedure steps

Step	Action
1	Enable system logging to a PCMCIA card: **`boot config flags logging`** If the logging flag is not set to true, the entries are stored in memory.
2	Configure the logfile parameters: **`boot config logfile <64-500> <500-16384> <10-90>`**

--End--

Variable definitions

Use the data in the following table to help you use the **`boot config`** commands in this procedure.

Variable	Value
`flags logging`	Enables or disables logging to a PCMCIA card. The log file is named using an 8.3 (xxxxxxxx.sss) format. The first six characters of the file name contain the last three bytes of the chassis base MAC address. The next two characters specify the slot number of the SF/CPU that generated the logs. The last three characters denote the sequence number of the log file. Multiple sequence numbers are generated for the same chassis and same

Variable	Value
	slot, if the SF/CPU is replaced, reinserted, or if the maximum log file size is reached.
`logfile <64-500> <500-16384> <10-90>`	Configures the logfile parameters: • `<64-500>` specifies the minimum space used for the logfile from 64–500 KB. • `<500-16384>` specifies the minimum space used for the logfile from 500–16384 KB. • `<10-90>` specifies the maximum percentage of PCMCIA space used for the logfile from 10–90%.

Starting system message logging to a PCMCIA card

Begin or stop logging system messages to the PCMCIA card.

When you remove the PCMCIA card from the primary SF/CPU, a trap is generated and system logging continues only in DRAM.
Start system message logging to a PCMCIA card by performing this procedure.

CAUTION
Risk of data loss
Before removing the PCMCIA card from your primary SF/CPU, you must stop the logging of system messages. Failure to do so can corrupt the file system on the PCMCIA card and cause your log file to be permanently lost.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Start logging system messages on the PCMCIA card: `log logToPCMCIA`
2	Stop logging: `no log logToPCMCIA`

--End--

Configuring system message control

Configure system message control to suppress duplicate error messages on the console, and to determine the action to take if they occur. Configure system message control by performing this procedure.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Configure system message control action: `sys msg-control action <suppress-msg|send-trap|both>`
2	Configure the maximum number of messages: `sys msg-control max-msg-num <2-500>`
3	Configure the interval: `sys msg-control control-interval control-interval <1-30>`
4	Enable message control: `sys msg-control`

--End--

Variable definitions

Use the information in the following table to complete the **`sys msg-control`** command.

Variable	Value
`action <suppress-msg|send-trap|both>`	Configures the message control action.
`control-interval <1-30>`	Configures the message control interval in minutes. The valid options are 1–30.
`max-msg-num <2-500>`	Configures the number of occurrences of a message after which the control action happens. To set the maximum number of occurrences, enter a value from 2–500.

Extending system message control

Use the force message control option to extend the message control feature functionality to the software and hardware log messages.

To enable the message control feature, you must specify an action, control interval, and maximum message number. After enabling the feature, the log messages, which get repeated and cross the maximum message number in the control interval, trigger the force message feature. You can either suppress the message or send a trap notification, or both.
Extend system message control by performing this procedure.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Configure the force message control option: `sys force-msg <WORD 4-4>`
	--End--

Variable definitions

Use the information in the following table to help you use this command.

Variable	Value
`<WORD 4-4>`	Used to add a forced message control pattern, where `<WORD 4-4>` is a string of 4 characters. You can add a four-byte pattern into the force-msg table. The software and the hardware log messages that use the first four bytes matching one of the patterns in the force-msg table undergo the configured message control action. You can specify up to 32 different patterns in the force-msg table. This includes a wildcard pattern (****) as well. Upon specifying the wildcard pattern, all messages undergo message control.

Configuring NNCLI logging

When enabled, NNCLI logging keeps track of all command line interface commands executed on the switch. Use NNCLI logging for fault management purposes.
Configure NNCLI logging by performing this procedure.

Prerequisites

- Access Global Configuration mode.

Procedure steps

Step	Action
1	Enable NNCLI logging: **clilog enable**
2	Change the maximum file size used for NNCLI logs: **clilog maxfilesize <64 to 256000>**
3	Ensure that the configuration is correct: **show clilog**
4	View the NNCLI log: **show clilog file [tail] [grep <WORD 1-256>]**

--End--

Variable definitions

Use the information in the following table to help you use the **clilog** commands.

Variable	Value
enable	Enables NNCLI logging. To disable, use the **no clilog enable** command.
maxfilesize <64 to 256000>	Specifies the maximum file size of the log file in KB.

Use the information in the following table to help you use the **show clilog file** commands.

Variable	Value
tail	Shows the last results first.
grep <WORD 1-256>	Performs a string search in the log file. **<WORD 1-256>** is the string, of up to 256 characters in length, to match.

Link state change control using Enterprise Device Manager

Use the procedure in this chapter to detect and control link flapping.

Controlling link state changes using Enterprise Device Manager

Use the following procedure to configure link flap detection to control link state changes on a physical port.

Procedure steps

Step	Action
1	In the navigation pane, open the following folders:**Edit, Diagnostics**.
2	Double-click **General**.
3	Click the **Link Flap** tab.
4	Configure the tab as required.
5	Click **Apply**.

--End--

Variable definitions

Use the information in the following table to help configure Link Flap Detect.

Variable	Value
AutoPortDownEnable	Enables or disables Link Flap Detect. If you enable Link Flap Detect, the switch monitors the number of times a port goes down during a designated interval. If the number of drops exceeds a specified limit, the system forces the port out-of-service.

Variable	Value
SendTrap	Specifies that a trap is sent if the port is forced out-of-service.
Frequency	Specifies the number of times the port can go down. The default is 10.
Interval	Specifies the interval (in seconds) between port failures. The default is 60.

Link state change control using CLI

Use the procedures in this chapter to detect and control link flapping.

Controlling link state changes using CLI

Use the following procedure to configure link flap detection to control state changes on a physical port.

Procedure steps

Step	Action
1	Configure the interval for link state changes: **config sys link-flap-detect interval <interval>**
2	Configure the number of changes allowed during the interval: **config sys link-flap-detect frequency <frequency>**
	--End--

Variable definitions

Use the data in the following table to use the **config sys link-flap-detect** command.

Variable	Value
auto-port-down <enable|disable>	Activates or disables automatic disabling of the port if the link-flap threshold is exceeded. The default is **enable**.
frequency <frequency>	Configures the number of changes that are allowed during the time specified by the **interval** command. • **frequency** is expressed in a range from 1–9999. The default value is 10.
info	Shows the link-flap-detect settings.

Variable	Value
interval <interval>	Configures the link-flap-detect interval in seconds. • **interval** is expressed in a range from 2–600. The default value is 60.
send-trap <enable|disable>	Activates or disables sending traps. The default is **enable**.

Example of controlling link state changes

Procedure steps

Step	Action

1	Enable automatic disabling of the port:
	`ERS-8606:5# config sys link-flap-detect`
	`ERS-8606:5/config/sys/link-flap-detect# auto-port-down enable`

2	Configure the link-flap-detect interval: `ERS-8606:5/config/sys/link-flap-detect# interval 20`
3	Enable sending traps: `ERS-8606:5/config/sys/link-flap-detect# send-trap enable`

4	Show the current configuration:
	`ERS-8606:5/config/sys/link-flap-detect# info`
	`Auto Port Down : enable`
	`Send Trap : enable`
	`Interval : 20`
	`Frequency : 10`
	`ERS-8606:5/config/sys/link-flap-detect#`
	You can display the same information with the **show sys link-flap-detect general-info** command.

--End--

Link state change control using NNCLI

Use the procedures in this chapter to detect and control link flapping.

Controlling link state changes using NNCLI

Use the following procedure to configure link flap detection to control state changes on a physical port.

Prerequisites

- You must log on to the Nortel Networks command line interface (NNCLI) Global Configuration mode.

Procedure steps

Step	Action
1	Configure the interval for link state changes: **`link-flap-detect interval <interval>`**
2	Configure the number of changes allowed during the interval: **`link-flap-detect frequency <frequency>`**
3	Enable automatic port disabling: **`link-flap-detect auto-port-down`**
4	Enable sending a trap: **`link-flap-detect send-trap`**
	--End--

Variable definitions

Use the data in the following table to use the **`link-flap-detect`** command.

Variable	Value
<auto-port-down>	Automatically disables the port if state changes exceed the link-flap threshold. By default, auto-port-down is disabled. Use the **no** operator to remove this configuration. To set this option to the default value, use the **default** operator with the command.
<frequency>	Configures the number of changes that are allowed during the time specified by the **interval** command. • **frequency** is from 1–9999. The default is 10. To set this option to the default value, use the **default** operator with the command.
<interval>	Configures the link-flap-detect interval in seconds. • **interval** is expressed in a range from 2–600. The default value is 60. To set this option to the default value, use the **default** operator with the command.
<send-trap>	Activates traps transmission. The default setting is activated. Use the **no** operator to remove this configuration. To set this option to the default value, use the **default** operator with the command.

Example of controlling link state changes

Procedure steps

Step	Action
1	Enable automatic disabling of the port: `ERS-8606:5(config)# link-flap-detect auto-port-down`
2	Configure the link-flap-detect interval: `ERS-8606:5(config)# link-flap-detect interval 20`
3	Enable sending traps: `ERS-8606:5(config)# link-flap-detect send-trap`

--End--

RMON alarm variables

This reference section describes Remote Monitoring (RMON) alarm variables.

RMON alarm variables are divided into three categories.

- Security
- Errors
- Traffic

Each category can have subcategories.

RMON alarm reference

The following table lists the alarm variable categories, subcategories where applicable, variable names, and provides a brief description of each variable.

Table 5
RMON alarm variables

Category	Subcategory	Variable	Definition
Security		rcCliNumAccessViolations.0	The number of command line interface (CLI) access violations detected by the system.
		rcWebNumAccessBlocks.0	The number of accesses the Web server blocked.
		snmpInBadCommunityNames.0	The total number of Simple Network Management Protocol (SNMP) messages delivered to the SNMP entity that represented an SNMP operation not allowed by the SNMP community named in the message.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
Errors	**Interface**	ifInDiscards	The number of inbound packets discarded even though no errors were detected to prevent the packets being deliverable to a higher-layer protocol. One possible reason for discarding a packet is to free buffer space.
		ifInErrors	For packet-oriented interfaces, the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol. For character-oriented or fixed-length interfaces, the number of inbound transmission units that contained errors, preventing them from being deliverable to a higher-layer protocol.
		ifOutDiscards	The number of outbound packets discarded even though no errors were detected to prevent the packets being transmitted. One possible reason for discarding such a packet is to free buffer space.
		ifOutErrors	For packet-oriented interfaces, the number of outbound packets that were not transmitted because of errors. For character-oriented or fixed-length interfaces, the number of outbound transmission units that were not transmitted because of errors.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
	Ethernet	dot3StatsAlignmentErrors	A count of frames received on a particular interface that are not an integral number of octets in length and do not pass the Frame Check Sequence (FCS) check. The count represented by an instance of this object increments when the alignmentError status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions exist are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.
		dot3StatsFCSErrors	A count of frames received on a particular interface that are an integral number of octets in length but do not pass the FCS check. The count represented by an instance of this object increments when the frameCheckError status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions occur are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		dot3StatsSingleCollisionFrames	A count of successfully transmitted frames on a particular interface where transmission is inhibited by exactly one collision. A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts object, the ifOutMulticastPkts object, or the ifOutBroadcastPkts object, and is not counted by the corresponding instance of the dot3StatsMultipleCollisionFrames object.
		dot3StatsMultipleCollisionFrames	A count of successfully transmitted frames on a particular interface where transmission is inhibited by more than one collision. A frame that is counted by an instance of this object is also counted by the corresponding instance of either the ifOutUcastPkts, ifOutMulticastPkts, or ifOutBroadcastPkts object, and is not counted by the corresponding instance of the dot3StatsSingleCollisionFrames object.
		dot3StatsSQETestErrors	A count of times that the SQE TEST ERROR message is generated by the PLS sublayer for a particular interface. The SQE TEST ERROR message is defined in section 7.2.2.2.4 of ANSI/IEEE 802.3-1985 and its generation is described in section 7.2.4.6 of the same document.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		dot3StatsDeferredTransmissions	A count of frames for which the first transmission attempt on a particular interface is delayed because the medium is busy. The count represented by an instance of this object does not include frames involved in collisions.
		dot3StatsLateCollisions	The number of times that a collision is detected on a particular interface later than 512 bit-times into the transmission of a packet; 512 bit-times corresponds to 51.2 microseconds on a 10 Mb/s system. A (late) collision included in a count represented by an instance of this object is also considered as a (generic) collision for purposes of other collision-related statistics.
		dot3StatsExcessiveCollisions	A count of frames where the transmission on a particular interface fails due to excessive collisions.
		dot3StatsInternalMacTransmitErrors	A count of frames where the transmission on a particular interface fails due to an internal MAC sublayer transmit error. A frame is counted by an instance of this object only if it is not counted by the corresponding instance of either the dot3StatsLateCollisions object, the dot3StatsExcessiveCollisions object, or the dot3StatsCarrierSenseErrors object. The precise meaning of the count represented by an instance of this object is

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
			implementation specific. In particular, an instance of this object can represent a count of transmission errors on a particular interface that are not otherwise counted.
		dot3StatsCarrierSenseErrors	The number of times the carrier sense condition was lost or never asserted when the switch attempted to transmit a frame on a particular interface. The count represented by an instance of this object increments at most once for each transmission attempt, even if the carrier sense condition fluctuates during a transmission attempt.
		dot3StatsFrameTooLongs	A count of frames received on a particular interface that exceeds the maximum permitted frame size. The count represented by an instance of this object increments when the frameTooLong status is returned by the MAC service to the LLC (or other MAC user). Received frames for which multiple error conditions obtained are, according to the conventions of IEEE 802.3 Layer Management, counted exclusively according to the error status presented to the LLC.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		dot3StatsInternalMacReceiveErrors	A count of frames where the transmission on a particular interface fails due to an internal MAC sublayer transmit error. A frame is counted by an instance of this object ony if it is not counted by the corresponding instance of either the dot3StatsLateCollisions object, the dot3StatsExcessiveCollisions object, or the dot3StatsCarrierSenseErrors object. The precise meaning of the count represented by an instance of this object is implementation-specific. In particular, an instance of this object can represent a count of transmission errors on a particular interface that are not otherwise counted.
	IP	ipInHdrErrors.0	The number of input datagrams discarded due to errors in the datagram IP headers, including bad checksums, version number mismatch, other format errors, time-to-live exceeded, and errors discovered in processing IP options.
		ipInDiscards.0	The number of discarded input IP datagrams where no problems were encountered to prevent continued processing. An example of why they were discarded can be lack of buffer space. This counter does not include any datagrams discarded while awaiting reassembly.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		ipOutDiscards.0	The number of output IP datagrams where no problems were encountered to prevent transmission to the destination, but that were discarded (for example, for lack of buffer space). This counter includes datagrams counted in ipForwDatagrams if packets meet this (discretionary) discard criterion.
		ipFragFails.0	The number of IP datagrams discarded because they needed to be fragmented at this entity but were not, for example, because the Don't Fragment flag was set.
		ipReasmFails.0	The number of failures detected by the IP reassembly algorithm (for whatever reason: timed out, errors, and so forth). This is not necessarily a count of discarded IP fragments because some algorithms (notably the algorithm in RFC 815) can lose track of the number of fragments by combining them as they are received.
		icmpInParmProbs.0	The number of ICMP In parameter problem messages received.
		icmpOutParmProbs.0	The number of ICMP Out parameter problem messages received.
	MLT	rcStatMltEtherAlignmentErrors	The number of frames received on an MLT that are not an integral number of octets in length, but do not pass the FCS check.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		rcStatMltEtherFCSErrors	The number of frames received on an MLT that are an integral number of octets in length, but do not pass the FCS check.
		rcStatMltEtherSingleCollFrames	The number of successfully transmitted frames on a particular MLT where transmission is inhibited by exactly one collision.
		rcStatMltEtherMultipleCollFrames	The number of successfully transmitted frames on a particular MLT where transmission is inhibited by more than one collision.
		rcStatMltEtherSQETestError	A count of times that the SQE TEST ERROR message is generated by the PLS sublayer for a particular MLT.
		rcStatMltEtherDeferredTransmiss	A count of frames where the first transmission attempt on a particular MLT is delayed because the medium is busy. The count represented by an instance of this object.
		rcStatMltEtherLateCollisions	The number of times that a late collision is detected on a particular MLT later than 512 bit-times into the transmission of a packet; 512-bit-times corresponds to 51.2-microseconds on a 10 Mb/s system.
		rcStatMltEtherExcessiveCollis	The number of times that excessive collisions are detected on a particular MLT later than 512 bit-times into the transmission of a packet; 512 bit-times corresponds to 51.2 microseconds on a 10-Mb/s system.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		rcStatMltEtherMacTransmitError	A count of frames where the transmission on a particular MLT fails due to an internal MAC sublayer transmit error. A frame is counted by an instance of this object only if it is not counted by the corresponding instance of either the LateCollisions object, the ExcessiveCollisions object, or the CarrierSenseErrors object.
		rcStatMltEtherCarrierSenseError	The number of times the carrier sense condition was lost or never asserted when attempting to transmit a frame on a particular MLT. The count represented by an instance of this object increments at most once for each transmission attempt, even if the carrier sense condition fluctuates during a transmission attempt.
		rcStatMltEtherFrameTooLong	A count of frames received on a particular MLT that exceeds the maximum permitted frame size. The count represented by an instance of this object increments when the frameTooLong status is returned by the MAC service to the LLC (or other MAC user).
		rcStatMltEtherMacReceiveError	A count of frames for which reception on a particular MLT fails due to an internal MAC sublayer receive error. A frame is counted by an instance of this object only if it is not counted by the corresponding instance of either the FrameTooLongs object, the AlignmentErrors object, or the FCSErrors object.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
	Other	rcTblArNoSpace	The number of entries not added to the address translation table due to lack of space.
		snmpInAsnParseErrs.0	The total number of ASN.1 or BER errors encountered by the SNMP protocol entity when it decodes received SNMP messages.
		rcStgPortInBadBpdus	The number of bad BPDUs received by this port.
		dot1dTpPortInDiscards	Count of valid frames received that were discarded (that is, filtered) by the forwarding process.
		rip2ifStatRcvBadPackets	The number of routes in valid RIP packets that were ignored for any reason.
		rip2ifStatRcvBadRoutes	The number of RIP response packets received by the RIP process that were subsequently discarded for any reason.
		rcStatOspfBufferAllocFailures.0	The number of times that OSPF failed to allocate buffers.
		rcStatOspfBufferFreeFailures.0	The number of times that OSPF failed to free buffers.
Traffic	**Interface**	ifInOctets	The total number of octets received on the interface, including framing characters.
		ifInMulticastPkts	The number of packets, delivered by this sublayer to a higher sublayer, that are addressed to a multicast address at this sublayer. For a MAC layer protocol, this number includes both Group and Functional addresses.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		ifInBroadcastPkts	The number of packets, delivered by this sublayer to a higher (sub) layer, that are addressed to a broadcast address at this sublayer.
		ifInUnkownProtos	For packet-oriented interfaces, the number of packets received through the interface that are discarded because of an unknown or unsupported protocol. For character-oriented or fixed-length interfaces that support protocol multiplexing, the number of transmission units received through the interface that are discarded because of an unknown or unsupported protocol. For any interface that does not support protocol multiplexing, this counter is always 0.
		ifOutOctets	The total number of octets transmitted from the interface, including framing characters.
		ifOutMulticastPkts	The total number of packets that higher-level protocols requested be transmitted, and that are addressed to a multicast address at this sublayer, including those that are discarded or not sent. For a MAC layer protocol, this includes both Group and Functional addresses.
		ifoutBroadcastPkts	The total number of packets that higher level protocols requested transmitted, and that were addressed to a broadcast address at this sublayer, including those discarded or not sent.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		ifLastChange	The value of sysUpTime at the time the interface entered its current operational state. If the current state was entered prior to the last reinitialization of the local network management subsystem, this object contains a value of zero.
	RmonEther Stats	etherStatsOctets	The total number of octets of data (including those in bad packets) received on the network (excluding framing bits but including FCS octets). Use this object as a reasonable estimate of Ethernet utilization. For greater precision, sample the etherStatsPkts and etherStatsOctets objects before and after a common interval.
		etherStatsPkts	The total number of packets (including bad packets, broadcast packets, and multicast packets) received.
		etherStatsBroadcastPkts	The total number of good packets received that are directed to the broadcast address. This number does not include multicast packets.
		etherStatsMulticastPkts	The total number of good packets received that are directed to a multicast address. This number does not include packets directed to the broadcast address.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		etherStatsCRCAlignErrors	The total number of packets received that had a length (excluding framing bits, but including FCS octets) of 64 to 1518 octets, inclusive, but had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error).
		etherStatsUndersizePkts	The total number of packets received that are less than 64 octets long (excluding framing bits, but including FCS octets) and were otherwise well formed.
		etherStatsOversizePkts	The total number of packets received that are longer than 1518 octets (excluding framing bits, but including FCS octets) and were otherwise well formed.
		etherStatsFragments	The total number of packets received that are less than 64 octets in length (excluding framing bits but including FCS octets) and had either a bad Frame Check Sequence (FCS) with an integral number of octets (FCS Error) or a bad FCS with a nonintegral number of octets (Alignment Error). It is entirely normal for etherStatsFragments to increment because it counts both runts (which are normal occurrences due to collisions) and noise hits.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		etherStatsCollisions	The best estimate of the total number of collisions on this Ethernet segment.
	IP	ipInReceives.0	All incoming IP packets.
		ipInAddrErrors.0	The number of bad IP destination addresses.
		ipForwDatagrams.0	IP packets forwarded.
		ipInUnknownProtos.0	Number of unsupported IP protocols.
		ipInDelivers.0	The number of IP In packets delivered.
		ipOutRequests.0	The total number of IP datagrams that local IP user protocols supplied to IP in request for transmission.
		ipOutNoRoutes.0	The number of IP datagrams discarded because no route was found to transmit to the destination.
		ipFragOKs.0	The number of IP datagrams successfully fragmented.
		ipFragCreates.0	The number of IP datagram fragments generated as a result of fragmentation.
		ipReasmReqds.0	The number of requests to reassemble fragments.
		ipReasmOKs.0	The number of fragments reassembled successfully.
	ICMP	IcmpInSrcQuenchs.0	The number of ICMP Source Quench messages received.
		icmpInRedirects.0	The number of ICMP redirect messages.
		icmpInEchos.0	The number of ICMP Echo requests messages received.
		icmpInEchosReps.0	The number of ICMP Echo reply messages received.
		icmpInTimeStamps.0	The number of ICMP timestamp request messages received.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		icmpInTimeStampsReps.0	The number of ICMP timestamp reply messages received.
		icmpInAddrMasks.0	The number of ICMP mask request messages reviewed.
		icmpInAddrMasksReps.0	The number of ICMP mask reply messages reviewed.
		icmpInDestUnreachs.0	The number of ICMP destinations unreachable messages received.
		icmpInTimeExcds.0	The number of ICMP Time Exceeded messages received.
		icmpOutSrcQuenchs.0	The number of ICMP Source Quench messages sent.
		icmpOutRedirects.0	The number of ICMP redirect messages sent.
		icmpOutEchos.0	The number of ICMP Echo request messages sent.
		icmpOutEchosReps.0	The number of ICMP Echo reply messages sent.
		icmpOutTimeStamps.0	The number of ICMP Timestamp request messages sent.
		icmpOutTimeStampsReps.0	The number of ICMP Timestamp reply messages sent.
		icmpOutAddrMasks.0	The number of ICMP Address mask messages sent.
		icmpOutAddrMasksReps.0	The number of ICMP Address mask reply messages sent.
		icmpOutDestUnreachs.0	The number of ICMP destination unreachable messages sent.
		icmpOutTimeExcds.0	The number of ICMP time exceeded messages sent.
	Snmp	snmpInPkts.0	The total number of messages delivered to the SNMP entity from the transport service.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		snmpOutPkts.0	The total number of SNMP messages passed from the SNMP protocol entity to the transport service.
		snmpInBadVersions.0	The total number of SNMP messages delivered to the SNMP protocol entity that were intended for an unsupported SNMP version.
		snmpInBadCommunityUses.0	The total number of SNMP messages delivered to the SNMP protocol entity that represented an SNMP operation that was not allowed by the SNMP community named in the message.
		snmpInTooBigs.0	The total number of SNMP PDUs delivered to the SNMP protocol entity and for which the value of the error-status field is tooBig.
		snmpInNoSuchNames.0	The total number of SNMP PDUs delivered to the SNMP protocol entity and for which the value of the error-status field is noSuchName.
		snmpInBadValues. 0	The total number of SNMP PDUs received that were generated by the SNMP protocol entity and for which the value of the error-status field is badValue.
		snmpInReadOnlys.0	The total number of valid SNMP PDUs delivered to the SNMP protocol entity and for which the value of the error-status field is readOnly. It is a protocol error to generate an SNMP PDU that contains the value readOnly in the error-status field; as such, this object is provided as a means of detecting incorrect implementations of the SNMP.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		snmpInGenErrs.0	The total number of SNMP PDUs delivered to the SNMP protocol entity and for which the value of the error-status field is genErr.
		snmpInTotalReqVars.0	The total number of MIB objects retrieved successfully by the SNMP protocol entity as the result of receiving valid SNMP Get-Request and Get-Next PDUs.
		snmpInTotalSetVars.0	The total number of MIB objects altered successfully by the SNMP protocol entity as the result of receiving valid SNMP Set-Request PDUs.
		snmpInGetRequests.0	The total number of SNMP Get-Request PDUs accepted and processed by the SNMP protocol entity.
		snmpInGetNexts.0	The total number of SNMP Get-Next PDUs accepted and processed by the SNMP protocol entity.
		snmpInSetRequests.0	The total number of SNMP Set-Request PDUs accepted and processed by the SNMP protocol entity.
		snmpInGetResponses.0	The total number of SNMP Get-Response PDUs accepted and processed by the SNMP protocol entity.
		snmpInTraps.0	The total number of SNMP Trap PDUs accepted and processed by the SNMP protocol entity.
		snmpOutTooBigs.0	The total number of SNMP PDUs generated by the SNMP protocol entity and for which the value of the error-status field is tooBig.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		snmpOutNoSuchNames.0	The total number of SNMP PDUs generated by the SNMP protocol entity and for which the value of the error-status field is noSuchName.
		snmpOutBadValues.0	The total number of SNMP PDUs sent that were generated by the SNMP protocol entity and for which the value of the error-status field is badValue.
		snmpOutGenErrs.0	The total number of SNMP PDUs generated by the SNMP protocol entity and for which the value of the error-status field is genErr.
		snmpOutGetRequests.0	The total number of SNMP Get-Request PDUs generated by the SNMP protocol entity.
		snmpOutGetNexts.0	The total number of SNMP Get-Next PDUs generated by the SNMP protocol entity.
		snmpOutSetRequests.0	The total number of SNMP Set-Request PDUs generated by the SNMP protocol entity.
		snmpOutGetResponses.0	The total number of SNMP Get-Response PDUs generated by the SNMP protocol entity.
		snmpOutTraps.0	The total number of SNMP Trap PDUs generated by the SNMP protocol entity.
	Bridge	rcStgTimeSinceTopologyChange	The time (in hundredths of a second) since the last topology change was detected by the bridge entity.
		rcStgTopChanges	The total number of topology changes detected by this bridge since the management entity was last reset or initialized.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		rcStgMaxAge	The maximum age of Spanning Tree Protocol information learned from the network on any port before it is discarded, in hundredths of a second. This is the actual value that this bridge is currently using.
		rcStgPortForwardTransitions	The number of times this port transitioned from the Learning state to the Forwarding state.
		rcStgPortInConfigBpdus	The number of Config BPDUs received by this port.
		rcStgPortInTcnBpdus	The number of Topology Change Notification BPDUs received by this port.
		rcStgPortOutConfigBpdus	The number of Config BPDUs transmitted by this port.
		rcStgPortOutTcnBpdus	The number of Topology Change Notification BPDUs transmitted by this port.
		dot1dTpPortInFrames	The number of frames received by this port from its segment. A frame received on the interface corresponding to this port is counted by this object only if it is for a protocol being processed by the local bridging function, including bridge management frames.
		dot1dTpPortOutFrames	The number of frames transmitted by this port to its segment. A frame transmitted on the interface corresponding to this port is counted by this object if and only if it is for a protocol processed by the local bridging function, including bridge management frames.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		dot1dTpLearnedEntryDiscards.0	The total number of Forwarding Database entries learned but discarded due to a lack of space to store them in the Forwarding Database. If this counter increases, it indicates that the forwarding database is regularly becoming full (a condition that has negative performance effects on the subnetwork). If this counter has a significant value but does not increase, it indicates that the problem occurred but is not persistent.
	Utilization	rcSysCpuUtil.0	Percentage of SF/CPU utilization.
		rcSysSwitchFabricUtil.0	Percentage of switching fabric utilization.
		rcSysBufferUtil.0	Buffer utilization as a percentage of the total amount of buffer space in the system. A high value indicates congestion.
		rcSysNVRamUsed.0	Nonvolatile RAM (NVRAM) in use in kilobytes.
		rcSysLastChange.0	Last management-initiated configuration change since sysUpTime.
		rcSysLastVlanChange.0	Last management-initiated VLAN configuration change since sysUpTime.
		rcSysLastSaveToNVRam.0	SysUpTime of the last time the NVRAM on the SF/CPU board was written to.
		rcSysLastSaveToStandbyNVRam .0	SysUpTime of the last time the standby NVRAM (on the backup SF/CPU board) was written to.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
	RIP	rip2GlobalRoute Changes.0	The number of changes made to the IP Route database by RIP.
		rip2GlobalQueries.0	The number of responses sent to RIP queries from other systems.
		rip2ifStatSentUpdates	The number of triggered RIP updates actually sent on this interface.
	OSPF	ospfExternLSACount.0	The number of external (LSA type 5) link-state advertisements in the link-state database.
		ospfOriginateNewLSAs.0	The number of new link-state advertisements that have originated. The number increments each time the router originates a new LSA.
		ospfrxNewLSAs.0	The number of link-state advertisements received determined to be new installations.
		ospfSpfRuns	Indicates the number of SPF calculations performed by OSPF.
		ospfAreaBdrRtrCount	The total number of area border routers reachable within this area.
		ospfASBdrRtrCount	The total number of autonomous system border routers reachable within this area.
		ospfAreaLSACount	The total number of link-state advertisements in this area link state database.
		ospfIfState	This signifies a change in the state of an OSPF virtual interface.
		ospfIfEvents	The number of times this OSPF interface changed the state or an error occurred.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		ospfVirtIfState	The number of times this OSPF interface.
		ospfVirtIfEvents	The number of state changes or error events on this virtual link.
		ospfVirtNbrState	The state of the Virtual Neighbor Relationship.
		ospfVirtNbrEvents	The number of times this virtual link changed the state or an error occurred.
	Igmp	igmpInterfaceWrongVersions	The number of queries received whose IGMP version does not match. IGMP requires that all routers on the LAN are configured to run the same version of IGMP.
		igmpInterfaceJoins	The number of times a group membership was added on this interface.
		igmpInterfaceLeaves	The number of times a group membership was deleted on this interface.
	MLT	rcStatMltIfExtnIfInMulticastPkts	The total number of multicast packets delivered to this MLT interface.
		rcStatMltIfExtnIfInBroadcastPkts	The total number of broadcast packets delivered to this MLT Interface.
		rcStatMltIfExtnIfOutMulticastPkts	The total number of MLT interface multicast packets delivered to this MLT interface.
		rcStatMltIfExtnIfOutBroadcastPkts	The total number of MLT interface broadcast packets delivered to this MLT interface.
		rcStatMltIfExtnIfHCInOctets	The total number of octets received on this MLT interface including framing characters detected by the high-count (64-bit) register.

Table 5
RMON alarm variables (cont'd.)

Category	Subcategory	Variable	Definition
		rcStatMltIfExtnIfHCInUcastPkts	The number of packets delivered by this MLT interface to a higher MLT that were not addressed to a multicast or broadcast address as detected by the high-count (64-bit) register.
		rcStatMltIfExtnIfHCInMulticastPkt	The total number of multicast packets delivered to this MLT interface detected by the high-count (64-bit) register.
		rcStatMltIfExtnIfHCInBroadcastPkt	The total number of broadcast packets delivered to this MLT interface detected by the high-count (64-bit) register.
		rcStatMltIfExtnIfHCOutOctets	The total number of octets transmitted from the MLT interface, including framing characters.
		rcStatMltIfExtnIfHCOutUcastPkts	The number of packets transmitted by this MLT interface to a higher MLT that were not addressed to a multicast or broadcast address as detected by the high-count (64-bit) register.
		rcStatMltIfExtnIfHCOutMulticast	The total number of packets that higher-level protocols requested be transmitted, and that were addressed to a multicast address at this sublayer, including those that were discarded or not sent registered by the high-count (64-bit) register.
		rcStatMltIfExtnIfHCOutBroadcast	The total number of packets that higher-level protocols requested be transmitted, and that were addressed to a broadcast address at this sublayer, including those that were discarded or not sent registered by the high-count (64-bit) register.

Nortel Ethernet Routing Switch 8600

Fault Management

Release: 7.0
Publication: NN46205-705
Document revision: 02.01
Document release date: 21 December 2009

To provide feedback or to report a problem in this document, go to www.nortel.com/documentfeedback.

www.nortel.com