JUNIPER NETWORKS

Junos® OS

SNMP MIBs and Traps Monitoring and Troubleshooting Guide for Security Devices

Release
15.1X49-D10

Modified: 2015-06-18

Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Junos® OS SNMP MIBs and Traps Monitoring and Troubleshooting Guide for Security Devices
15.1X49-D10

The information in this document is current as of the date on the title page.

YEAR 2000 NOTICE

Juniper Networks hardware and software products are Year 2000 compliant. Junos OS has no known time-related limitations through the year 2038. However, the NTP application is known to have some difficulty in the year 2036.

END USER LICENSE AGREEMENT

The Juniper Networks product that is the subject of this technical documentation consists of (or is intended for use with) Juniper Networks software. Use of such software is subject to the terms and conditions of the End User License Agreement ("EULA") posted at http://www.juniper.net/support/eula.html. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

To obtain the most current version of all Juniper Networks® technical documentation, see the product documentation page on the Juniper Networks website at http://www.juniper.net/techpubs/.

If the information in the latest release notes differs from the information in the documentation, follow the product Release Notes.

Juniper Networks Books publishes books by Juniper Networks engineers and subject matter experts. These books go beyond the technical documentation to explore the nuances of network architecture, deployment, and administration. The current list can be viewed at http://www.juniper.net/books.

Supported Platforms

For the features described in this document, the following platforms are supported:

- vSRX
- LN Series
- SRX Series

Using the Examples in This Manual

If you want to use the examples in this manual, you can use the **load merge** or the **load merge relative** command. These commands cause the software to merge the incoming configuration into the current candidate configuration. The example does not become active until you commit the candidate configuration.

If the example configuration contains the top level of the hierarchy (or multiple hierarchies), the example is a *full example*. In this case, use the **load merge** command.

If the example configuration does not start at the top level of the hierarchy, the example is a *snippet*. In this case, use the **load merge relative** command. These procedures are described in the following sections.

Merging a Full Example

To merge a full example, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration example into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following configuration to a file and name the file **ex-script.conf**. Copy the **ex-script.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   system {
     scripts {
       commit {
         file ex-script.xsl;
       }
     }
   }
   interfaces {
     fxp0 {
       disable;
       unit 0 {
         family inet {
           address 10.0.0.1/24;
         }
       }
     }
   }
   ```

2. Merge the contents of the file into your routing platform configuration by issuing the **load merge** configuration mode command:

   ```
   [edit]
   user@host# load merge /var/tmp/ex-script.conf
   load complete
   ```

Merging a Snippet

To merge a snippet, follow these steps:

1. From the HTML or PDF version of the manual, copy a configuration snippet into a text file, save the file with a name, and copy the file to a directory on your routing platform.

   For example, copy the following snippet to a file and name the file **ex-script-snippet.conf**. Copy the **ex-script-snippet.conf** file to the **/var/tmp** directory on your routing platform.

   ```
   commit {
     file ex-script-snippet.xsl; }
   ```

2. Move to the hierarchy level that is relevant for this snippet by issuing the following configuration mode command:

```
[edit]
user@host# edit system scripts
[edit system scripts]
```

3. Merge the contents of the file into your routing platform configuration by issuing the **load merge relative** configuration mode command:

```
[edit system scripts]
user@host# load merge relative /var/tmp/ex-script-snippet.conf
load complete
```

For more information about the **load** command, see the *CLI User Guide*.

Documentation Conventions

Table 1 on page xvii defines notice icons used in this guide.

Table 1: Notice Icons

Icon	Meaning	Description
	Informational note	Indicates important features or instructions.
	Caution	Indicates a situation that might result in loss of data or hardware damage.
	Warning	Alerts you to the risk of personal injury or death.
	Laser warning	Alerts you to the risk of personal injury from a laser.
	Tip	Indicates helpful information.
	Best practice	Alerts you to a recommended use or implementation.

Table 2 on page xviii defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Encloses optional keywords or variables.	**stub <default-metric** *metric***>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	**community name members [** *community-ids* **]**

Indention and braces ({ })	Identifies a level in the configuration hierarchy.	[edit]
		routing-options {
		  static {
		  route default {
		  nexthop *address*;
		  retain;
		  }
		  }
		}

; (semicolon)	Identifies a leaf statement at a configuration hierarchy level.	

GUI Conventions

Table 2: Text and Syntax Conventions *(continued)*

Convention	Description	Examples
Bold text like this	Represents graphical user interface (GUI) items you click or select.	• In the Logical Interfaces box, select **All Interfaces**. • To cancel the configuration, click **Cancel**.
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can provide feedback by using either of the following methods:

- Online feedback rating system—On any page at the Juniper Networks Technical Documentation site at http://www.juniper.net/techpubs/index.html, simply click the stars to rate the content, and use the pop-up form to provide us with information about your experience. Alternately, you can use the online feedback form at https://www.juniper.net/cgi-bin/docbugreport/.
- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active J-Care or Partner Support Service support contract, or are covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

- JTAC policies—For a complete understanding of our JTAC procedures and policies, review the *JTAC User Guide* located at http://www.juniper.net/us/en/local/pdf/resource-guides/7100059-en.pdf.
- Product warranties—For product warranty information, visit http://www.juniper.net/support/warranty/.
- JTAC hours of operation—The JTAC centers have resources available 24 hours a day, 7 days a week, 365 days a year.

Self-Help Online Tools and Resources

For quick and easy problem resolution, Juniper Networks has designed an online self-service portal called the Customer Support Center (CSC) that provides you with the following features:

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: http://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

To verify service entitlement by product serial number, use our Serial Number Entitlement (SNE) Tool: https://tools.juniper.net/SerialNumberEntitlementSearch/

Opening a Case with JTAC

You can open a case with JTAC on the Web or by telephone.

- Use the Case Management tool in the CSC at http://www.juniper.net/cm/.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

For international or direct-dial options in countries without toll-free numbers, see http://www.juniper.net/support/requesting-support.html.

PART 1

Overview

CHAPTER 1

Introduction to Device Management

Understanding Device Management Functions in Junos OS

Supported Platforms LN Series, SRX Series

After you have installed a device into your network, you need to manage the device within your network. Device management can be divided into five tasks:

- Fault management—Monitor the device; detect and fix faults.
- Configuration management—Configure device attributes.
- Accounting management—Collect statistics for accounting purposes.
- Performance management—Monitor and adjust device performance.
- Security management—Control device access and authenticate users.

The Junos OS network management features work in conjunction with an operations support system (OSS) to manage the devices within the network. Junos OS can assist you in performing these management tasks, as described in Table 3 on page 4.

Table 3: Device Management Features in Junos OS

Task	Junos OS Feature
Fault management	Monitor and see faults using: • Operational mode commands—For more information about operational mode commands, see the *CLI User Guide*. • SNMP MIBs—For more information about SNMP MIBs supported by Junos OS, see "Standard SNMP MIBs Supported by Junos OS" on page 13 and "Juniper Networks Enterprise-Specific MIBs" on page 32. • Standard SNMP traps—For more information about standard SNMP traps, see the "Standard SNMP Traps Supported on Devices Running Junos OS" on page 95. • Enterprise-specific SNMP traps—For more information about enterprise-specific traps, see "Juniper Networks Enterprise-Specific SNMP Traps" on page 80. • System log messages—For more information about how to configure system log messages, see *System Log Monitoring and Troubleshooting Guide for Security Devices*.
Configuration management	• Configure device attributes using the command-line interface (CLI). For more information about configuring the device using the CLI, see the *CLI User Guide*. • Configuration Management MIB—For more information about the Configuration Management MIB, see the *Configuration Management MIB*.
Accounting management	Perform the following accounting-related tasks: • Collect statistics for interfaces, firewall filters, destination classes, source classes, and the Routing Engine. For more information about collecting statistics, see *Accounting Options Configuration*. • Use interface-specific traffic statistics and other counters, available in the Standard Interfaces MIB, Juniper Networks enterprise-specific extensions to the Interfaces MIB, and media-specific MIBs, such as the enterprise-specific ATM MIB. • Count packets as part of a firewall filter. For more information about firewall filter policies, see "Juniper Networks Enterprise-Specific MIBs" on page 32 and the *Junos OS Routing Protocols Library for Security Devices*.
Performance management	Monitor performance in the following ways: • Use operational mode commands. For more information about monitoring performance using operational mode commands, see the *CLI User Guide*. • Use firewall filters. For more information about performance monitoring using firewall filters, see the *Junos OS Routing Protocols Library for Security Devices*.

Table 3: Device Management Features in Junos OS *(continued)*

Task	Junos OS Feature
Security management	Assure security in your network in the following ways: • Control access to the router and authenticate users. For more information about access control and user authentication, see the *Junos OS User Authentication Library for Security Devices*. • Control access to the router using SNMPv3 and SNMP over IPv6. For more information, see "Configuring the Local Engine ID" on page 156 and "Tracing SNMP Activity on a Device Running Junos OS" on page 197.

Related Documentation

- Understanding the SNMP Implementation in Junos OS on page 9
- *Accounting Options Overview*

Understanding the Integrated Local Management Interface

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

The Integrated Local Management Interface (ILMI) provides a mechanism for Asynchronous Transfer Mode (ATM)-attached devices, such as hosts, routers, and ATM switches, to transfer management information. ILMI provides bidirectional exchange of management information between two ATM interfaces across a physical connection. ILMI information is exchanged over a direct encapsulation of SNMP version 1 (RFC 1157, *A Simple Network Management Protocol*) over ATM Adaptation Layer 5 (AAL5) using a virtual path identifier/virtual channel identifier (VPI/VCI) value (VPI=0, VCI=16).

Junos OS supports only two ILMI MIB variables: **atmfMYIPNmAddress** and **atmfPortMyIfname**. For ATM1 and ATM2 intelligent queuing (IQ) interfaces, you can configure ILMI to communicate directly with an attached ATM switch to enable querying of the switch's IP address and port number.

For more information about the ILMI MIB, see the ATM Forum at http://www.atmforum.com/.

Related Documentation

- Understanding Device Management Functions in Junos OS on page 3

PART 2

Network Monitoring Using SNMP

CHAPTER 2

SNMP MIBs Overview

Understanding the SNMP Implementation in Junos OS

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

SNMP enables the monitoring of network devices from a central location. This topic provides an overview of SNMP and describes how SNMP is implemented in the Junos OS.

This topic includes the following sections:

SNMP Architecture

The SNMP agent exchanges network management information with SNMP manager software running on a network management system (NMS), or host. The agent responds to requests for information and actions from the manager. The agent also controls access to the agent's MIB, the collection of objects that can be viewed or changed by the SNMP manager.

The SNMP manager collects information about network connectivity, activity, and events by polling managed devices.

Communication between the agent and the manager occurs in one of the following forms:

- **Get**, **GetBulk**, and **GetNext** requests—The manager requests information from the agent; the agent returns the information in a **Get** response message.
- **Set** requests—The manager changes the value of a MIB object controlled by the agent; the agent indicates status in a **Set** response message.
- **Traps** notification—The agent sends traps to notify the manager of significant events that occur on the network device.

This topic contains the following sections:

SNMP MIBs

A MIB is a hierarchy of information used to define managed objects in a network device. The MIB structure is based on a tree structure, which defines a grouping of objects into related sets. Each object in the MIB is associated with an object identifier (OID), which names the object. The "leaf" in the tree structure is the actual managed object instance, which represents a resource, event, or activity that occurs in your network device.

MIBs are either standard or enterprise-specific. Standard MIBs are created by the Internet Engineering Task Force (IETF) and documented in various RFCs. Depending on the vendor, many standard MIBs are delivered with the NMS software. You can also download the standard MIBs from the IETF website, www.ietf.org, and compile them into your NMS, if necessary.

For a list of standard supported MIBs, see "Standard SNMP MIBs Supported by Junos OS" on page 13.

Enterprise-specific MIBs are developed and supported by a specific equipment manufacturer. If your network contains devices that have enterprise-specific MIBs, you must obtain them from the manufacturer and compile them into your network management software.

For a list of Juniper Networks enterprise-specific supported MIBs, see "Juniper Networks Enterprise-Specific MIBs" on page 32.

SNMP Traps and Informs

Routers can send notifications to SNMP managers when significant events occur on a network device, most often errors or failures. SNMP notifications can be sent as traps or inform requests. SNMP traps are unconfirmed notifications. SNMP informs are confirmed notifications.

SNMP traps are defined in either standard or enterprise-specific MIBs. Standard traps are created by the IETF and documented in various RFCs. The standard traps are compiled into the network management software. You can also download the standard traps from the IETF website, www.ietf.org.

For more information about standard traps supported by the Junos OS, see "Standard SNMP Traps Supported on Devices Running Junos OS" on page 95.

Enterprise-specific traps are developed and supported by a specific equipment manufacturer. If your network contains devices that have enterprise-specific traps, you must obtain them from the manufacturer and compile them into your network management software.

For more information about enterprise-specific traps supported by the Junos OS, see "Juniper Networks Enterprise-Specific SNMP Traps" on page 80. For information about

system logging severity levels for SNMP traps, see "System Logging Severity Levels for SNMP Traps" on page 11.

With traps, the receiver does not send any acknowledgment when it receives a trap, and the sender cannot determine if the trap was received. To increase reliability, SNMP informs are supported in SNMPv3. An SNMP manager that receives an inform acknowledges the message with a response. For information about SNMP informs, see "Configuring SNMP Informs" on page 157.

SNMP Trap Queuing

The Junos OS supports trap queuing to ensure that traps are not lost because of temporary unavailability of routes. Two types of queues, destination queues and a throttle queue, are formed to ensure delivery of traps and to control the trap traffic.

The Junos OS forms a destination queue when a trap to a particular destination is returned because the host is not reachable, and adds the subsequent traps to the same destination to the queue. The Junos OS checks for availability of routes every 30 seconds and sends the traps from the destination queue in a round-robin fashion. If the trap delivery fails, the trap is added back to the queue, and the delivery attempt counter and the next delivery attempt timer for the queue are reset. Subsequent attempts occur at progressive intervals of 1 minute, 2 minutes, 4 minutes, and 8 minutes. The maximum delay between the attempts is 8 minutes, and the maximum number of attempts is 10. After 10 unsuccessful attempts, the destination queue and all the traps in the queue are deleted.

The Junos OS also has a throttle mechanism to control the number of traps (**throttle threshold**; default value of 500 traps) sent during a particular time period (**throttle interval**; default of 5 seconds) and to ensure consistency in trap traffic, especially when a large number of traps are generated because of interface status changes. The throttle interval period begins when the first trap arrives at the throttle. All traps within the trap threshold are processed, and the traps beyond the threshold limit are queued. The maximum size of trap queues (that is, the throttle queue and the destination queue combined) is 40,000 traps. However, on EX Series switches, the maximum size of the trap queue is 1000 traps. The maximum size of any one queue is 20,000 traps for devices other than EX Series switches. On EX Series switches, the maximum size of one queue is 500 traps. If a trap is sent from a destination queue when the throttle queue has exceeded the maximum size, the trap is added back to the top of the destination queue, and all subsequent attempts from the destination queue are stopped for a 30-second period, after which the destination queue restarts sending the traps.

NOTE: Users cannot configure the Junos OS for trap queuing. Users cannot view any information about trap queues except what is available in the syslog.

System Logging Severity Levels for SNMP Traps

For some traps, when a trap condition occurs, regardless of whether the SNMP agent sends a trap to an NMS, the trap is logged if the system logging is configured to log an event with that system logging severity level. For more information about system logging severity levels, see the *System Log Monitoring and Troubleshooting Guide for Security Devices*.

For more information about system logging severity levels for standard traps, see "Standard SNMP Version 1 Traps" on page 96 and "Standard SNMP Version 2 Traps" on page 99. For more information about system logging severity levels for enterprise-specific traps, see "Juniper Networks Enterprise-Specific SNMP Version 1 Traps" on page 81 and "Juniper Networks Enterprise-Specific SNMP Version 2 Traps" on page 88.

Junos OS SNMP Agent Features

The Junos OS SNMP agent software consists of an SNMP master agent that delegates all SNMP requests to subagents. Each subagent is responsible for the support of a specific set of MIBs.

The Junos OS supports the following versions of SNMP:

- SNMPv1—The initial implementation of SNMP that defines the architecture and framework for SNMP.
- SNMPv2c—The revised protocol, with improvements to performance and manager-to-manager communications. Specifically, SNMPv2c implements community strings, which act as passwords when determining who, what, and how the SNMP clients can access the data in the SNMP agent. The community string is contained in SNMP **Get**, **GetBulk**, **GetNext**, and **Set** requests. The agent might require a different community string for **Get**, **GetBulk**, and **GetNext** requests (**read-only** access) than it does for **Set** requests (**read-write** access).
- SNMPv3—The most up-to-date protocol focuses on security. SNMPv3 defines a security model, user-based security model (USM), and a view-based access control model (VACM). SNMPv3 USM provides data integrity, data origin authentication, message replay protection, and protection against disclosure of the message payload. SNMPv3 VACM provides access control to determine whether a specific type of access (read or write) to the management information is allowed.

In addition, the Junos OS SNMP agent software accepts IPv4 and IPv6 addresses for transport over IPv4 and IPv6. For IPv6, the Junos OS supports the following features:

- SNMP data over IPv6 networks
- IPv6-specific MIB data
- SNMP agents for IPv6

Related Documentation

- *System Log Monitoring and Troubleshooting Guide for Security Devices*
- SNMPv3 Overview on page 138
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

CHAPTER 3

SNMP MIBs and Traps Supported by Junos OS

Standard SNMP MIBs Supported by Junos OS

Supported Platforms ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, T Series

Table 4 on page 14 contains the list of standard SNMP MIBs and RFCs that are supported on various devices running Junos OS. RFCs can be found at http://www.ietf.org .

NOTE: In this table, a value of 1 in any of the platform columns (M, T, MX, EX, and SRX) denotes that the corresponding MIB is supported on that particular platform, and a value of 0 denotes that the MIB is not supported on the platform.

Table 4: Standard MIBs Supported on Devices Running Junos OS

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
IEEE 802.1ab section 12.1, *Link Layer Discovery Protocol (LLDP) MIB* EX Series implementation of LLDP MIB supports both IPv4 and IPv6 configuration.	0	0	0	0	1	0	0	0	0
IEEE, 802.3ad, *Aggregation of Multiple Link Segments* Supported tables and objects: • **dot3adAggPortTable**, **dot3adAggPortListTable**, **dot3adAggTable**, and **dot3adAggPortStatsTable** NOTE: EX Series switches do not support the **dot3adAggPortTable** and **dot3adAggPortStatsTable**. • **dot3adAggPortDebugTable** (only **dot3adAggPortDebugRxState**, **dot3adAggPortDebugMuxState**, **dot3adAggPortDebugActorSyncTransitionCount**, **dot3adAggPortDebugPartnerSyncTransitionCount**, **dot3adAggPortDebugActorChangeCount**, and **dot3adAggPortDebugPartnerChangeCount**) NOTE: EX Series switches do not support the **dot3adAggPortDebugTable**. • **dot3adTablesLastChanged**	0	1	1	1	1	1	1	1	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms							
						SRX		
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End
IEEE, 802.1ag, *Connectivity Fault Management* Supported tables and objects: • **dot1agCfmMdTableNextIndex** • **dot1agCfmMdTable** (except **dot1agCfmMdMhfIdPermission**) • **dot1agCfmMaNetTable** • **dot1agCfmMaMepListTable** • **dot1agCfmDefaultMdDefLevel** • **dot1agCfmDefaultMdDefMhfCreation** • **dot1agCfmMepTable** (except **dot1agCfmMepLbrBadMsdu**, **dot1agCfmMepTransmitLbmVlanPriority**, **dot1agCfmMepTransmitLbmVlanDropEnable**, **dot1agCfmMepTransmitLtmFlags**, **dot1agCfmMepPbbTeCanReportPbbTePresence**, **dot1agCfmMepPbbTeTrafficMismatchDefect**, **dot1agCfmMepPbbTransmitLbmLtmReverseVid**, **dot1agCfmMepPbbTeMismatchAlarm**, **dot1agCfmMepPbbTeLocalMismatchDefect**, and **dot1agCfmMepPbbTeMismatchSinceReset**) • **dot1agCfmLtrTable** (except **dot1agCfmLtrChassisIdSubtype**, **dot1agCfmLtrChassisId**, **dot1agCfmLtrManAddressDomain**, **dot1agCfmLtrManAddress**, **dot1agCfmLtrIngressPortIdSubtype**, **dot1agCfmLtrIngressPortId**, **dot1agCfmLtrEgressPortIdSubtype**, **dot1agCfmLtrEgressPortId**, and **dot1agCfmLtrOrganizationSpecificTlv**) • **dot1agCfmMepDbTable** (except **dot1agCfmMebDbChassisIdSubtype**, **dot1agCfmMebDbChassisId**, **dot1agCfmMebDbManAddressDomain**, and **dot1agCfmMebDbManAddress**)	0	0	0	1	0	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
IEEE, 802.1ap, *Management Information Base (MIB) definitions for VLAN Bridges* Supported tables and objects: • **ieee8021CfmStackTable** • **ieee8021CfmVlanTable** • **ieee8021CfmDefaultMdTable** (except **ieee8021CfmDefaultMdIdPermission**) • **ieee8021CfmMaCompTable** (except **ieee8021CfmMaCompIdPermission**)	0	0	0	1	0	0	0	0	
RFC 1155, *Structure and Identification of Management Information for TCP/IP-based Internets*	1	1	1	1	1	1	1	1	1
RFC 1157, *A Simple Network Management Protocol (SNMP)*	1	1	1	1	1	1	1	1	1
RFC 1195, *Use of OSI IS-IS for Routing in TCP/IP and Dual Environments* (only the objects **isisSystem**, **isisMANAreaAddr**, **isisAreaAddr**, **isisSysProtSupp**, **isisSummAddr**, **isisCirc**, **isisCircLevel**, **isisPacketCount**, **isisISAdj**, **isisISAdjAreaAddr**, **isisAdjIPAddr**, **isisISAdjProtSupp**, **isisRa**, and **isisIPRA** are supported)	1	1	1	1	1	1	1	1	1
RFC 1212, *Concise MIB Definitions*	1	1	1	1	1	1	0	0	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	

RFC 1213, *Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II*. Junos OS supports the following areas:
• MIB II and its SNMP version 2 derivatives, including:
  ◦ Statistics counters
  ◦ IP, except for **ipRouteTable**, which has been replaced by **ipCidrRouteTable** (RFC 2096, *IP Forwarding Table MIB*)
  ◦ SNMP management
  ◦ Interface management
• SNMPv1 **Get**, **GetNext** requests, and version 2 **GetBulk** request
• Junos OS-specific secured access list
• Master configuration keywords
• Reconfigurations upon SIGHUP
	1	1	1	1	1	1	0	0	1

RFC 1215, *A Convention for Defining Traps for use with the SNMP* (only MIB II SNMP version 1 traps and version 2 notifications)	1	1	1	1	1	1	0	0	1
RFC 1406, *Definitions of Managed Objects for the DS1 and E1 Interface Types* (T1 MIB is supported)	1	1	1	0	0	0	1	0	0
RFC 1407, *Definitions of Managed Objects for the DS3/E3 Interface Type* (T3 MIB is supported)	0	1	1	0	0	0	0	0	0
RFC 1471, *Definitions of Managed Objects for the Link Control Protocol of the Point-to-Point Protocol* (only pppLink group is supported. The pppLink group consists of the **pppLcp1** object and the tables **pppLinkStatustable** and **pppLinkConfigTable**).	0	1	0	1	0	1	0	0	0
RFC 1657, *Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP-4) using SMIv2*	1	1	1	1	1	0	0	0	0
RFC 1695, *Definitions of Managed Objects for ATM Management Version 8.0 Using SMIv2*	1	1	1	0	0	1	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 1850, *OSPF Version 2 Management Information Base* (except for the **ospfOriginateNewLsas** and **ospfRxNewLsas** objects, the Host Table, and the traps **ospfOriginateLSA**, **ospfLsdbOverflow**, and **ospfLsdbApproachingOverflow**)	1	1	1	1	1	1	1	0	0
RFC 1901, *Introduction to Community-based SNMPv2*	1	1	1	1	1	1	1	1	1
RFC 2011, *SNMPv2 Management Information Base for the Internet Protocol Using SMIv2*	1	1	1	1	1	1	0	0	0
RFC 2012, *SNMPv2 Management Information Base for the Transmission Control Protocol Using SMIv2*	1	1	1	1	1	1	1	0	1
RFC 2013, *SNMPv2 Management Information Base for the User Datagram Protocol Using SMIv2*	1	1	1	1	1	1	1	0	1
RFC 2024, *Definitions of Managed Objects for Data Link Switching Using SMIv2* (except for the **dlswInterface** and **dlswSdlc** object groups; the **dlswDirLocateMacTable**, **dlswDirNBTable**, and **dlswDirLocateNBTable** tables; the **dlswCircuitDiscReasonLocal** and **dlswCircuitDiscReasonRemote** tabular objects; and the **dlswDirMacCacheNextIndex** and **dlswDirNBCacheNextIndex** scalar objects; read-only access)	0	1	1	1	0	0	0	0	0
RFC 2096, *IP Forwarding Table MIB* (The **ipCidrRouteTable** has been extended to include the tunnel name when the next hop is through an RSVP-signaled LSP.) NOTE: RFC 2096 has been replaced by RFC 4292. However, Junos OS currently supports both RFC 2096 and RFC 4292.	1	1	1	1	1	1	0	0	1
RFC 2115, *Management Information Base for Frame Relay DTEs Using SMIv2* (**frDlcmiTable** only; **frCircuitTable** and **frErrTable** are not supported)	0	1	1	1	0	0	1	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 2233, *The Interfaces Group MIB Using SMIv2* NOTE: RFC 2233 has been replaced by RFC 2863, IF MIB. However, Junos OS supports both RFC 2233 and RFC 2863.	1	1	1	1	1	1	1	0	1
RFC 2287, *Definitions of System-Level Managed Objects for Applications* (only the objects **sysApplInstallPkgTable**, **sysApplInstallElmtTable**, **sysApplElmtRunTable**, and **sysApplMapTable**)	1	1	1	1	1	1	1	0	1
RFC 2465, *Management Information Base for IP Version 6: Textual Conventions and General Group* (except for IPv6 interface statistics)	1	1	1	1	0	1	1	0	0
RFC 2495, *Definitions of Managed Objects for the DS1, E1, DS2, and E2 Interface Types* (except for **dsx1FarEndConfigTable**, **dsx1FarEndCurrentTable**, **dsx1FarEndIntervalTable**, **dsx1FarEndTotalTable**, and **dsx1FracTable**)	1	1	1	0	0	0	1	0	0
RFC 2515, *Definitions of Managed Objects for ATM Management* (except **atmVpCrossConnectTable**, **atmVcCrossConnectTable**, and **aal5VccTable**)	1	1	1	0	0	0	0	0	0
RFC 2570, *Introduction to Version 3 of the Internet-standard Network Management Framework*	1	1	1	1	1	1	0	0	1
RFC 2571, *An Architecture for Describing SNMP Management Frameworks* (read-only access) NOTE: RFC 2571 has been replaced by RFC 3411. However, Junos OS supports both RFC 2571 and RFC 3411.	1	1	1	1	1	1	1	0	1
RFC 2572, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)* (read-only access) NOTE: RFC 2572 has been replaced by RFC 3412. However, Junos OS supports both RFC 2572 and RFC 3412.	1	1	1	1	1	1	1	0	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 2576, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework* NOTE: RFC 2576 has been replaced by RFC 3584. However, Junos OS supports both RFC 2576 and RFC 3584.	1	1	1	1	1	1	1	0	1
RFC 2578, *Structure of Management Information Version 2 (SMIv2)*	1	1	1	1	1	1	0	0	1
RFC 2579, *Textual Conventions for SMIv2*	1	1	1	1	1	1	0	0	1
RFC 2580, *Conformance Statements for SMIv2*	1	1	1	1	1	1	0	0	1
RFC 2662, *Definitions of Managed Objects for ADSL Lines* (All MIB tables, objects, and traps are applicable for the ADSL ATU-R agent.)	0	1	1	1	0	0	1	0	0
RFC 2665, *Definitions of Managed Objects for the Ethernet-like Interface Types* NOTE: For M, T and MX Series, the SNMP counters do not count the Ethernet header and frame check sequence (FCS). Therefore, the Ethernet header bytes and the FCS bytes are not included in the following four OIDs: • ifInOctets • ifOutOctets • ifHCInOctets • ifHCOutOctets However, the EX switches adhere to RFC 2665. NOTE: The list of managed objects specified in RFC 2665 has been updated by RFC 3635 by including information useful for the management of 10 Gigabit per second Ethernet interfaces.	1	1	1	1	1	1	1	0	1
RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol* (except row creation, the **Set** operation, and the object **vrrpStatsPacketLengthErrors**)	1	1	1	1	1	1	1	0	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 2790, *Host Resources MIB* • Only the **hrStorageTable**. The file systems **/**, **/config**, **/var**, and **/tmp** always return the same index number. When SNMP restarts, the index numbers for the remaining file systems might change. • Only the objects of the **hrSystem** and **hrSWInstalled** groups.	1	1	1	1	1	1	1	0	1
RFC 2819, *Remote Network Monitoring Management Information Base* • **etherStatsTable** (for Ethernet interfaces only),**alarmTable**, **eventTable**, and **logTable** are supported on all devices running Junos OS. • **historyControlTable** and **etherHistoryTable** (except **etherHistoryUtilization** object) are supported only on EX Series switches.	1	1	1	1	1	1	1	0	1
RFC 2863, *The Interfaces Group MIB* NOTE: RFC 2863 replaces RFC 2233. However, Junos OS supports both RFC 2233 and RFC 2863.	1	1	1	1	1	1	0	0	1
RFC 2864, *The Inverted Stack Table Extension to the Interfaces Group MIB*	0	1	1	1	0	1	0	0	1
RFC 2922, *The Physical Topology (PTOPO) MIB* Supported objects: **ptopoConnDiscAlgorithm, ptopoConnAgentNetAddrType, ptopoConnAgentNetAddr, ptopoConnMultiMacSASeen, ptopoConnMultiNetSASeen, ptopoConnIsStatic, ptopoConnLastVerifyTime, ptopoConnRowStatus**	0	0	0	0	1	0	1	0	1
RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations* (only the objects **pingCtlTable**, **pingResultsTable**, **pingProbeHistoryTable**, **pingMaxConcurrentRequests**, **traceRouteCtlTable**, **traceRouteResultsTable**, **traceRouteProbeHistoryTable**, and **traceRouteHopsTable**)	1	1	1	1	1	1	1	0	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 2932, *IPv4 Multicast Routing MIB*	1	1	1	1	1	1	1	0	1
RFC 2934, *Protocol Independent Multicast MIB for IPv4* NOTE: In Junos OS, RFC 2934 is implemented based on a draft version, *pimmib.mib*, of the now standard RFC. Support for the **pimNeighborLoss** trap was added in Release 11.4.	1	1	1	1	1	1	1	0	0
RFC 2981, *Event MIB*	1	1	1	1	0	1	0	0	0
RFC 3014, *Notification Log MIB*	1	1	1	1	0	1	0	0	0
RFC 3019, *IP Version 6 Management Information Base for The Multicast Listener Discovery Protocol*	0	1	1	1	0	1	0	0	1
RFC 3410 *Introduction and Applicability Statements for Internet-Standard Management Framework*	1	1	1	1	1	1	0	0	1
RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks* NOTE: RFC 3411 replaces RFC 2571. However, Junos OS supports both RFC 3411 and RFC 2571.	1	1	1	1	1	1	0	0	1
RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)* NOTE: RFC 3412 replaces RFC 2572. However, Junos OS supports both RFC 3412 and RFC 2572.	1	1	1	1	1	1	0	0	1
RFC 3413, *Simple Network Management Protocol (SNMP) Applications* (except for the Proxy MIB)	1	1	1	1	1	1	1	0	1
RFC 3414, *User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*	1	1	1	1	1	1	0	0	1
RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*	1	1	1	1	1	1	0	0	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)* NOTE: RFC 3416 replaces RFC 1905, which was supported in earlier versions of Junos OS.	1	1	1	1	1	1	0	0	1
RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)*	1	1	1	1	1	1	1	0	1
RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)* NOTE: RFC 3418 replaces RFC 1907, which was supported in earlier versions of Junos OS.	1	1	1	1	1	1	0	0	1
RFC 3498, *Definitions of Managed Objects for Synchronous Optical Network (SONET) Linear Automatic Protection Switching (APS) Architectures* (implemented under the Juniper Networks enterprise branch [**jnxExperiment**])	0	1	1	0	0	0	0	0	0
RFC 3584 *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*	1	1	1	1	1	1	0	0	1
RFC 3591 *Managed Objects for the Optical Interface Type* **optIfOTMnTable** (except **optIfOTMnOpticalReach**, **optIfOTMnInterfaceType**, and **optIfOTMnOrder**), **optIfOChConfigTable** (except **optIfOChDirectionality** and **optIfOChCurrentStatus**), **optIfOTUkConfigTable** (except **optIfOTUkTraceIdentifierAccepted**, **optIfOTUkTIMDetMode**, **optIfOTUkTIMActEnabled**, **optIfOTUkTraceIdentifierTransmitted**, **optIfOTUkDEGThr**, **optIfOTUkDEGM**, **optIfOTUkSinkAdaptActive**, and **optIfOTUkSourceAdaptActive**), and **optIfODUkConfigTable** (except **optIfODUkPositionSeqCurrentSize** and **optIfODUkTtpPresent**)	0	1	1	0	0	0	0	0	0
RFC 3592, *Definitions of Managed Objects for the Synchronous Optical Network/Synchronous Digital Hierarchy (SONET/SDH) Interface Type*	0	1	1	1	0	0	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 3621, *Power Ethernet MIB*	0	0	0	0	1	0	0	0	0
RFC 3635, *Definitions of Managed Objects for the Ethernet-like Interface Types* (except **dot3StatsRateControlAbility** and **dot3StatsRateControlStatus** in **dot3StatsEntry** table) NOTE: The values of the following objects in **dot3HCStatsEntry** table will be always zero for both 32-bit counters and 64-bit counters: • **dot3HCStatsSymbolErrors** • **dotHCStatsInternalMacTransmitErrors**	0	0	0	1	0	0	0	0	0
RFC 3637, *Definitions of Managed Objects for the Ethernet WAN Interface Sublayer* (except **etherWisDeviceTable**, **etherWisSectionCurrentTable**, and **etherWisFarEndPathCurrentTable**)	0	1	1	1	0	1	0	0	0
RFC 3811, *Definitions of Textual Conventions (TCs) for Multiprotocol Label Switching (MPLS) Management*	1	1	1	1	0	1	1	0	0
RFC 3812, *Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)* (read only access) • MPLS tunnels as interfaces are not supported. • The following objects in the **TunnelResource** table are not supported: **mplsTunnelResourceMeanRate**, **mplsTunnelResourceMaxBurstSize**, **mplsTunnelResourceMeanBurstSize**, **mplsTunnelResourceExBurstSize**, **mplsTunnelResourceWeight**. • **mplsTunnelPerfTable** and **mplsTunnelCRLDPResTable** are not supported. • **mplsTunnelCHopTable** is supported on ingress routers only. NOTE: The branch used by the proprietary LDP MIB (**ldpmib.mib**) conflicts with RFC 3812. **ldpmib.mib** has been deprecated and replaced by **jnx-mpls-ldp.mib**.	1	1	1	1	0	1	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 3813, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base (MIB)* (read-only access). **mplsInterfacePerfTable**, **mplsInSegmentPerfTable**, **mplsOutSegmentPerfTable**, **mplsInSegmentMapTable**, **mplsXCUp**, and **mplsXCDown** are not supported.	1	1	1	1	0	1	1	0	0
RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model*	1	1	1	1	1	1	0	0	1
RFC 3877, *Alarm Management Information Base* except: • Junos OS does not support the **alarmActiveStatsTable.** • Traps that do not conform to the alarm model are not supported. However, these traps can be redefined to conform to the alarm model.	0	0	0	1	0	0	0	0	
RFC 3896, *Definitions of Managed Objects for the DS3/E3 Interface Type* (except **dsx3FarEndConfigTable**, **dsx3FarEndCurrentTable**, **dsx3FarEndIntervalTable**, **dsx3FarEndTotalTable**, and **dsx3FracTable**)	0	1	1	0	0	0	0	0	0
RFC 4087, *IP Tunnel MIB*—Describes MIB objects in the following tables for managing tunnels of any type over IPv4 and IPv6 networks: • **tunnelIfTable**—Provides information about the tunnels known to a router. • **tunnelInetConfigTable**—Assists dynamic creation of tunnels and provides mapping from end-point addresses to the current interface index value. NOTE: Junos OS supports **MAX-ACCESS** of read-only for all the MIB objects in **tunnelIfTable** and **tunnelInetConfigTable** tables.	0	1	1	1	0	0	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 4133, *Entity MIB*—Supports tables and objects except: • **entityLogicalGroup** table • **entPhysicalMfgDate** and **entPhysicalUris** objects in **entityPhysical2Group** table • **entLPMappingTable** and **entPhysicalContainsTable** in **entityMappingGroup** table • **entityNotofications Group** table NOTE: Supported only on MX240, MX480, and MX960 routers.	0	0	0	1	0	0	0	0	
RFC 4188, *Definitions of Managed Objects for Bridges*—Supports 802.1D STP(1998). Supports only the following subtrees and objects: • **dot1dStp** subtree is supported on MX Series 3D Universal Edge Routers . • **dot1dTpFdbAddress**, **dot1dTpFdbPort**, and **dot1dTpFdbStatus** objects from the **dot1dTpFdbTable** of the **dot1dTp** subtree are supported on EX Series Ethernet Switches. NOTE: **dot1dTpLearnedEntryDiscards** and **dot1dTpAgingTime** objects are supported on M and T Series routers.	0	0	0	1	1	0	0	0	
RFC 4268, *Entity State MIB*—Junos OS supports all objects and tables. NOTE: Supported only on MX240, MX480, and MX960 routers.	0	0	0	1	0	0	0	0	0
RFC 4273, *Definitions of Managed Objects for BGP-4* (only **jnxBgpM2PrefixInPrefixes**, **jnxBgpM2PrefixInPrefixesAccepted**, and **jnxBgpM2PrefixInPrefixesRejected** objects)	1	1	1	1	1	0	0	0	1

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 4292, *IP Forwarding MIB*— Describes a table and MIB objects for forwarding IP packets that are version independent: • **inetCidrRouteTable**—Provides the ability to display IP version-independent multipath CIDR routes and obsoletes the **ipCidrRouteTable** object. • **inetCidrRouteNumber**—Indicates the number of current routes and obsoletes the **ipCidrRouteNumber** object. • **inetCidrRouteDiscards**—Counts the number of valid routes that are discarded from **inetCidrRouteTable** and obsoletes the **ipCidrRouteDiscards** object. NOTE: Junos OS currently supports these MIB objects that will be deprecated in future releases: **ipCidrRouteTable**, **ipCidrRouteNumber**, and **ipCidrRouteDiscards**.	1	1	1	1	1	1	0	0	0
RFC 4293, *Management Information Base for the Internet Protocol (IP)* — Supports only the mandatory groups. For detailed information, see Standard IPv4/IPv6 MIBs.	0	0	0	1	1	0	0	0	0
RFC 4318, *Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol*—Supports 802.1w and 802.1t extensions for RSTP.	0	1	1	1	1	0	0	0	0
RFC 4363b, *Q-Bridge VLAN MIB*	0	0	0	1	1	0	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 4382 *MPLS/BGP Layer 3 Virtual Private Network (VPN) MIB* The Junos OS support for RFC 4382 includes the following scalar objects and tables: • **mplsL3VpnActiveVrfs** • **mplsL3VpnConfiguredVrfs** • **mplsL3VpnConnectedInterfaces** • **mplsL3VpnVrfConfMidRteThresh** • **mplsL3VpnVrfConfHighRteThresh** • **mplsL3VpnIfConfRowStatus** • **mplsL3VpnIllLblRcvThrsh** • **mplsL3VpnNotificationEnable** • **mplsL3VpnVrfConfMaxPossRts** • **mplsL3VpnVrfConfRteMxThrshTime** • **mplsL3VpnVrfOperStatus** • **mplsL3VpnVrfPerfCurrNumRoutes** • **mplsL3VpnVrfPerfTable** • **mplsVpnVrfRTTable** • **mplsL3VpnVrfSecIllegalLblVltns** • **mplsL3VpnVrfTable** NOTE: The **mplsL3VpnIfConfTable** has not been implemented in the MPLS/BGP Layer 3 Virtual Private Network (VPN) MIB, because of limited utility and difficulty in representing the **DistProtocol** bit accurately.	0	1	1	1	1	1	0	0	0
RFC 4444, *IS-IS MIB*	1	1	1	1	1	1	1	0	0
RFC 4668, *RADIUS Accounting Client Management Information Base (MIB) for IPv6* (read-only access)	0	0	0	1	0	0	0	0	0
RFC 4670, *RADIUS Accounting Client Management Information Base (MIB)* (read-only access)	0	0	0	1	0	0	0	0	0
RFC 4801, *Definitions of Textual Conventions for Generalized Multiprotocol Label Switching (GMPLS) Management Information Base (MIB)* (read-only access)	0	1	1	1	0	0	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 4802, *Generalized Multiprotocol Label Switching (GMPLS) Traffic Engineering (TE) Management Information Base (MIB)* (read-only access). **gmplsTunnelReversePerfTable**, **gmplsTeScalars**, **gmplsTunnelTable**, **gmplsTunnelARHopTable**, **gmplsTunnelCHopTable**, and **gmplsTunnelErrorTable** are not supported.)	0	1	1	1	0	0	0	0	0
RFC 4803, *Generalized Multiprotocol Label Switching (GMPLS) Label Switching Router (LSR) Management Information Base (MIB)* (read-only access). **gmplsLabelTable** and **gmplsOutsegmentTable** are not supported. NOTE: The tables in GMPLS TE (RFC 4802) and LSR (RFC 4803) MIBs are extensions of the corresponding tables from the MPLS TE (RFC 3812) and LSR (RFC 3813) MIBs and use the same index as the MPLS MIB tables.	0	1	1	1	0	0	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
RFC 5643, *Management Information Base for OSPFv3* NOTE: Junos OS support for this MIB is read-only. Junos OS does not support the following tables and objects defined in this MIB. • **ospfv3HostTable** • **ospfv3CfgNbrTable** • **ospfv3ExitOverflowInterval** • **ospfv3ReferenceBandwidth** • **ospfv3RestartSupport** • **ospfv3RestartInterval** • **ospfv3RestartStrictLsaChecking** • **ospfv3RestartStatus** • **ospfv3RestartAge** • **ospfv3RestartExitReason** • **ospfv3NotificationEnable** • **ospfv3StubRouterSupport** • **ospfv3StubRouterAdvertisement** • **ospfv3DiscontinuityTime** • **ospfv3RestartTime** • **ospfv3AreaNssaTranslatorRole** • **ospfv3AreaNssaTranslatorState** • **ospfv3AreaNssaTranslatorStabInterval** • **ospfv3AreaNssaTranslatorEvents** • **ospfv3AreaTEEnabled** • **ospfv3IfMetricValue** • **ospfv3IfDemandNbrProbe**	0	1	1	1	0	1	0	0	1
RFC 6527, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol Version 3 (VRRPv3)* (except row creation, the **Set** operation, and the objects **vrrpv3StatisticsRowDiscontinuityTime** and **vrrpv3StatisticsPacketLengthErrors**)	1	0	0	0	0	0	0	0	0
Internet Assigned Numbers Authority, *IANAiftype Textual Convention MIB* (referenced by RFC 2233, available at http://www.iana.org/assignments/ianaiftype-mib)	1	1	1	1	1	1	1	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
Internet draft draft-ietf-atommib-sonetaps-mib-10.txt, *Definitions of Managed Objects for SONET Linear APS Architectures* (as defined under the Juniper Networks enterprise branch [**jnxExperiment**] only)	0	1	1	1	0	0	0	0	0
Internet draft draft-ieft-bfd-mib-02.txt, *Bidirectional Forwarding Detection Management Information Base* (Represented by **mib-jnx-bfd-exp.txt** and implemented under the Juniper Networks enterprise branch [**jnxExperiment**]. Read only. Includes **bfdSessUp** and **bfdSessDown** traps. Does not support **bfdSessPerfTable** and **bfdSessMapTable**.)	1	1	1	1	1	0	0	0	1
Internet draft draft-ietf-l3vpn-mvpn-mib-03.txt, *MPLS/BGP Layer 3 VPN Multicast Management Information Base* (Implemented under the Juniper Networks enterprise branch [**jnxExperiment**]. OID for **jnxMvpnExperiment** is **.1.3.6.1.4.1.2636.5.12**. Read only. Includes **jnxMvpnNotifications** traps.)	0	1	1	0	0	0	0	0	
Internet draft draft-ietf-idmr-igmp-mib-13.txt, *Internet Group Management Protocol (IGMP) MIB*	0	1	1	1	1	1	0	0	1
Internet draft draft-reeder-snmpv3-usm-3desede-00.txt, *Extension to the User-Based Security Model (USM) to Support Triple-DES EDE in 'Outside' CBC Mode*	1	1	1	1	1	1	0	0	1
Internet draft draft-ietf-isis-wg-mib-07.txt, *Management Information Base for IS-IS* (only **isisISAdjTable**, **isisISAdjAreaAddrTable**, **isisISAdjIPAddrTable**, and **isisISAdjProtSuppTable)** NOTE: Replaced with RFC 4444, *IS-IS MIB* in Junos OS Release 11.3 and later.	1	1	1	1	1	1	1	0	0
Internet draft draft-ietf-ppvpn-mpls-vpn-mib-04.txt, *MPLS/BGP Virtual Private Network Management Information Base Using SMIv2* (only **mplsVpnScalars**, **mplsVpnVrfTable,** **mplsVpnPerTable**, and **mplsVpnVrfRouteTargetTable)**	0	1	1	1	0	1	0	0	0

Table 4: Standard MIBs Supported on Devices Running Junos OS *(continued)*

MIB/RFC	Platforms								
						SRX			
	ACX	M	T	MX	EX	Low-End	Mid-Range	High-End	
Internet draft draft-ietf-ospf-ospfv3-mib-11.txt, *Management Information Base for OSPFv3* (Represented by **mib-jnx-ospfv3mib.txt** and implemented under the Juniper Networks enterprise branch **{jnxExperiment}**. Support for **ospfv3NbrTable** only. Read only. Object names are prefixed by **jnx**. For example, **jnxOspfv3NbrTable**, **jnxOspfv3NbrAddressType**, and **jnxOspfv3NbrPriority**.)	0	1	1	1	0	1	0	0	1
Internet draft draft-ietf-idmr-pim-mib-09.txt, *Protocol Independent Multicast (PIM) MIB*	1	1	1	1	1	1	0	0	1
ESO Consortium MIB, which can be found at http://www.snmp.com/eso/ NOTE: The ESO Consortium MIB has been replaced by RFC 3826.	1	1	1	1	1	1	1	0	0
Internet Draft P2MP MPLS-TE MIB (draft-ietf-mpls-p2mp-te-mib-09.txt) (read-only access) (except **mplsTeP2mpTunnelBranchPerfTable**).	1	1	1	1	0	1	0	0	0

Related Documentation

- Juniper Networks Enterprise-Specific MIBs on page 32
- Loading MIB Files to a Network Management System on page 111

Juniper Networks Enterprise-Specific MIBs

Supported Platforms LN Series, SRX Series

The Junos OS supports the following enterprise-specific MIBs:

DHCP Objects MIB— Provides SNMP support (get and trap) for DHCP local server and relay configurations. It also provides support for bindings and leases tables, and for statistics. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-jdhcp.txt .

For more information, see *DHCP MIB*.

SNMP GetSNMP Traphttp://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-dom.txt .

For more information, see *Digital Optical Monitoring MIB*.

Power Supply Unit MIB—Enables monitoring and managing of the power supply on a device running the Junos OS. This MIB is currently supported only on EX Series Ethernet Switches. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-power-supply-unit.txt .

For more information, see *Power Supply Unit MIB*.

- AAA Objects MIB—Provides support for monitoring user authentication, authorization, and accounting through the RADIUS, LDAP, SecurID, and local authentication servers. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-user-aaa.txt

  For more information, see *AAA Objects MIB*.

- Access Authentication Objects MIB—Provides support for monitoring firewall authentication, including data about the users trying to access firewall-protected resources and the firewall authentication service itself. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-auth.txt.

  For more information, see *Access Authentication Objects MIB*.

- Alarm MIB—Provides support for alarms from the router. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis-alarm.txt .

  For more information, see *Alarm MIB*.

- DNS Objects MIB—Provides support for monitoring DNS proxy queries, requests, responses, and failures. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-dns.txt .

  For more information, see *DNS Objects MIB*.

- Firewall MIB—Provides support for monitoring firewall filter counters. Routers must have the Internet Processor II ASIC to perform firewall monitoring. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-firewall.txt .

  For more information, see *Firewall MIB*.

- Flow Collection Services MIB—Provides statistics on files, records, memory, FTP, and error states of a monitoring services interface. It also provides SNMP traps for unavailable destinations, unsuccessful file transfers, flow overloading, and memory overloading. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-coll.txt .

For more information, see *Flow Collection Services MIB*.

- Host Resources MIB—Extends the **hrStorageTable** object, providing a measure of the usage of each file system on the router in percentage format. Previously, the objects in the **hrStorageTable** measured the usage in allocation units—**hrStorageUsed** and **hrStorageAllocationUnits**—only. Using the percentage measurement, you can more easily monitor and apply thresholds on usage. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-hostresources.txt .

  For more information, see *Host Resources MIB*.

- IDP Objects MIB—Provides support for monitoring SNMP IDP queries, requests, responses, and failures. This MIB defines the key monitoring and threshold crossing trap support, IDP database update status and trap support, attack-related monitoring and trap support for all SRX Series devices. This MIB models IDP attributes specific to the appropriate Juniper Networks implementation. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-idp.txt .

  For more information, see *IDP MIB*.

- Interface MIB—Extends the standard **ifTable** (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-if-extensions.txt .

  For more information, see *Interface MIB*.

- Interface Accounting Forwarding Class MIB—Extends the Juniper Enterprise Interface MIB and provides support for monitoring statistcs data for interface accounting and IETF standardization. This MIB is currently supported by Junos OS for M Series and MX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-if-accounting.txt

  For more information, see *Interface Accounting Forwarding Class MIB*.

- IP Forward MIB—Extends the standard IP Forwarding Table MIB (RFC 2096) to include CIDR forwarding information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipforward.txt .

  For more information, see *IP Forward MIB*.

- IPsec Generic Flow Monitoring Object MIB—Based on **jnx-ipsec-monitor-mib**, this MIB provides support for monitoring IPsec and IPsec VPN management objects. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-flow-mon.txt .

  For more information, see *IPsec Generic Flow Monitoring Object MIB*.

- IPsec Monitoring MIB—Provides operational and statistical information related to the IPsec and IKE tunnels on Juniper Networks routers. For a downloadable version of this

MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-monitor-asp.txt .

For more information, see *IPsec Monitoring MIB*.

- IPsec VPN Objects MIB—Provides support for monitoring IPsec and IPsec VPN management objects for Juniper security product lines. This MIB is an extension of **jnx-ipsec-flow-mon.mib**. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-ipsec-vpn.txt .

  For more information, see *IPsec VPN Objects MIB*.

- IPv4 MIB—Provides additional Internet Protocol version 4 (IPv4) address information, supporting the assignment of identical IPv4 addresses to separate interfaces. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipv4.txt .

  For more information, see *IPv4 MIB*.

- IPv6 and ICMPv6 MIB—Provides IPv6 and Internet Control Message Protocol version 6 (ICMPv6) statistics. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipv6.txt .

  For more information, see *IPv6 MIB*.

- L2ALD MIB—Contains information about the Layer 2 Address Learning Daemon (L2ALD) and related traps, such as the routing instance MAC limit trap and the interface MAC limit trap. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-l2ald.txt .

  For more information, see *L2ALD MIB*.

- L2CP MIB—Provides information about Layer 2 Control Protocols (L2CP) based features on MX Series 3D Universal Edge Routers . Currently, Junos OS supports only the **jnxDot1dStpPortRootProtectEnabled**, **jnxDot1dStpPortRootProtectState**, and **jnxPortRootProtectStateChangeTrap** objects. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-l2cp-features.txt .

  For more information, see *L2CP MIB*.

- L2TP MIB—Provides information about Layer 2 Transport Protocol (L2TP) tunnels and sessions. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-l2tp.txt .

  For more information, see *L2TP MIB*.

- LDP MIB—Provides LDP statistics and defines LDP label-switched path (LSP) notifications. LDP traps support only IPv4 standards. For a downloadable version of

this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-ldp.txt .

For more information, see *LDP MIB*.

- License MIB—Extends SNMP support to licensing information, and introduces SNMP traps that alert users when the licenses are about to expire, expire, or when the total number of users exceeds the number specified in the license. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-license.txt .

  For more information, see *License MIB*.

- Logical Systems MIBs—Extend SNMP support to logical systems security profile through various MIBs defined under **jnxLsysSecurityProfile**. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-lsys-securityprofile.txt.

  For more information about logical systems MIBs and downloadable versions of the MIBs, see *Logical Systems MIB*.

- NAT Objects MIB—Provides support for monitoring network address translation (NAT). This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-nat.txt .

  For more information, see *NAT Objects MIB*.

- PPP MIB—Provides SNMP support for PPP-related information such as the type of authentication used, interface characteristics, status, and statistics. This MIB is currently supported only on M Series and MX Series routers. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-ppp.txt .

  For more information, see *PPP MIB*.

- PPPoE MIB—Provides SNMP support for PPPoE-related information such as the type of authentication used, interface characteristics, status, and statistics. This MIB is currently supported only on M Series and MX Series routers. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-pppoe.txt .

  For more information, see *PPPoE MIB*.

- Pseudowire TDM MIB—Extends the standard Pseudowire MIB, and contains information about configuration and statistics for specific pseudowire types. The enterprise-specific Pseudowire TDM MIB is the Juniper Networks implementation of the standard Managed Objects for TDM over Packet Switched Network MIB (draft-ietf-pwe3-tdm-mib-08.txt). For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-pwtdm.txt .

- SONET APS MIB—Monitors any SONET interface that participates in Automatic Protection Switching (APS). For a downloadable version of this MIB, see

http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-sonetaps.txt .

For more information, see *SONET APS MIB*.

- SONET/SDH Interface Management MIB—Monitors the current alarm for each SONET/SDH interface. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x45/topics/reference/mibs/mib-jnx-sonet.txt .

  For more information, see *SONET/SDH Interface Management MIB*.

- Source Class Usage MIB—Counts packets sent to customers by performing a lookup on the IP source address and the IP destination address. The Source Class Usage (SCU) MIB makes it possible to track traffic originating from specific prefixes on the provider core and destined for specific prefixes on the customer edge. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-scu.txt .

  For more information, see *Source Class Usage MIB*.

- SPU Monitoring MIB—Provides support for monitoring SPUs on all high-end SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-spu-monitoring.txt .

  For more information, see *SPU Monitoring Objects MIB*.

- Structure of Management Information MIB—Contains object identifiers (OIDs) for the security branch of the MIBs used in Junos OS for SRX Series devices, services, and traps. This MIB is currently supported by Junos OS for SRX Series devices only.

  Explains how the Juniper Networks enterprise-specific MIBs are structured. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-smi.txt .

  For more information, see *Structure of Management Information MIB*.

- Structure of Management Information MIB for EX Series Ethernet Switches—Defines a MIB branch for switching-related MIB definitions for the EX Series Ethernet Switches. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ex-smi.txt .

  For more information, see *EX Series SMI MIB*.

List of SRX100, SRX210, SRX220, SRX240, SRX550, and SRX650 Services Gateways Supported Enterprise-Specific MIBs

Supported Platforms LN Series, SRX100, SRX110, SRX210, SRX220, SRX240, SRX550, SRX650

Junos OS supports the following enterprise-specific MIBs:

- Structure of Management Information MIB—Contains object identifiers (OIDs) for the security branch of the MIBs used in Junos OS for SRX Series devices product, services and traps. This MIB is currently supported only by Junos OS for SRX Series devices. It also explains how the Juniper Networks enterprise-specific MIBs are structured. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-smi.txt. For more information, see *Structure of Management Information MIB*.

- Access Authentication Objects MIB—Provides support for monitoring firewall authentication, including data about the users trying to access firewall-protected resources and the firewall authentication service itself. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-auth.txt .

  For more information, see *Access Authentication Objects MIB*.

- Alarm MIB—Provides support for alarms from the router. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis-alarm.txt .

  For more information, see *Alarm MIB*.

- BGP4 V2 MIB—Contains objects used to monitor BGP peer-received prefix counters. It is based upon similar objects in the MIB documented in Internet draft draft-ietf-idr-bgp4-mibv2-03.txt, Definitions of Managed Objects for the Fourth Version of BGP (BGP-4), Second Version. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-bgpmib2.txt .

  For more information, see *BGP4 V2 MIB*.

- BFD MIB—Provides support for monitoring Bidirectional Forwarding Detection (BFD) sessions. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-bfd.txt.

  For more information, see *Bidirectional Forwarding Detection MIB*.

- Chassis MIB—Provides support for environmental monitoring (power supply state, board voltages, fans, temperatures, and air flow) and inventory support for the chassis, System Control Board (SCB), System and Switching Board (SSB), Switching and Forwarding Model (SFM), Flexible PIC Concentrators (FPCs), and PICs. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis.txt .

  For more information, see *Chassis MIBs*.

- Configuration Management MIB—Provides notification for configuration changes as SNMP traps. Each trap contains the time at which the configuration change was committed, the name of the user who made the change, and the method by which the change was made. A history of the last 32 configuration changes is kept in

jnxCmChgEventTable. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-cfgmgmt.txt .

For more information, see *Configuration Management MIB*.

- Ethernet MAC MIB—Monitors media access control (MAC) statistics on Gigabit Ethernet intelligent queuing (IQ) interfaces. It collects MAC statistics; for example, **inoctets**, **inframes**, **outoctets**, and **outframes** on each source MAC address and virtual LAN (VLAN) ID for each Ethernet port. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-mac.txt .

  For more information, see *Ethernet MAC MIB*.

- Event MIB—Defines a generic trap that can be generated using an op script or event policy. This MIB provides the ability to specify a system log string and raise a trap if that system log string is found. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-event.txt .

  For more information, see *Event MIB*.

- Firewall MIB—Provides support for monitoring firewall filter counters. Routers must have the Internet Processor II ASIC to perform firewall monitoring. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-firewall.txt .

  For more information, see *Firewall MIB*.

- Host Resources MIB—Extends the **hrStorageTable** object, providing a measure of the usage of each file system on the router in percentage. Previously, the objects in the **hrStorageTable** measured the usage in allocation units—**hrStorageUsed** and **hrStorageAllocationUnits**—only. Using the percentage measurement, you can more easily monitor and apply thresholds on usage. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-hostresources.txt .

  For more information, see *Host Resources MIB*.

- Interface MIB—Extends the standard ifTable (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-if-extensions.txt .

  For more information, see *Interface MIB*.

- IP Forward MIB—Extends the standard IP Forwarding Table MIB (RFC 2096) to include CIDR forwarding information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipforward.txt .

  For more information, see *IP Forward MIB*.

- IPsec Monitoring MIB—Provides operational and statistical information related to the IPsec and IKE tunnels on Juniper Networks routers. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-monitor-asp.txt .

  For more information, see *IPsec Monitoring MIB*.

- IPsec Generic Flow Monitoring Object MIB—Based on **jnx-ipsec-monitor-mib**, this MIB provides support for monitoring IPsec and IPsec VPN management objects. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-flow-mon.txt .

  For more information, see *IPsec Generic Flow Monitoring Object MIB*.

- IPv4 MIB—Provides additional Internet Protocol version 4 (IPv4) address information, supporting the assignment of identical IPv4 addresses to separate interfaces. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipv4.txt .

  For more information, see *IPv4 MIB*.

- License MIB—Extends SNMP support to licensing information, and introduces SNMP traps that alert users when the licenses are about to expire, expire, or when the total number of users exceeds the number specified in the license. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-license.txt .

  For more information, see *License MIB*.

- Network Address Translation (NAT) Objects MIB—Provides support for monitoring network address translation (NAT). This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-nat.txt .

  For more information, see *NAT Objects MIB*.

- Packet Forwarding Engine MIB—Provides notification statistics for Packet Forwarding Engines. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-pfe.txt .

  For more information, see *Packet Forwarding Engine MIB*.

- Ping MIB—Extends the standard Ping MIB control table (RFC 2925). Items in this MIB are created when entries are created in pingCtlTable of the Ping MIB. Each item is indexed exactly as it is in the Ping MIB. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ping.txt .

  For more information, see *PING MIB*.

- Policy Objects MIB—Provides support for monitoring the security policies that control the flow of traffic from one zone to another. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-policy.txt .

  For more information, see *Policy Objects MIB*.

- Reverse-Path-Forwarding MIB—Monitors statistics for traffic that is rejected because of reverse-path-forwarding (RPF) processing. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-rpf.txt .

  NOTE: The enterprise-specific RPF MIB is not supported on EX Series Ethernet Switches.

  For more information, see *Reverse Path Forwarding MIB*.

- RMON Events and Alarms MIB—Supports the Junos extensions to the standard Remote Monitoring (RMON) Events and Alarms MIB (RFC 2819). The extension augments alarmTable with additional information about each alarm. Two new traps are also defined to indicate when problems are encountered with an alarm. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-rmon.txt .

  For more information, see *RMON Events and Alarms MIB*.

- Security Interface Extension Objects MIB—Provides support for the security management of interfaces. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-if-ext.txt .

  For more information, see *Security Interface Extension Objects MIB*.

- SNMP IDP Objects MIB—Provides support for monitoring SNMP IDP queries, requests, responses, and failures. This MIB defines the key monitoring and threshold crossing trap support, IDP database update status and trap support, attack-related monitoring and trap support for SRX100, SRX210, SRX220, SRX240, SRX550, and SRX650 Services Gateways. This MIB models IDP attributes specific to the appropriate Juniper Networks implementation. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-idp.txt .

  For more information, see *SNMP IDP MIB*.

- System Log MIB—Enables notification of an SNMP trap-based application when an important system log message occurs. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-syslog.txt .

  For more information, see *System Log MIB*.

- Traceroute MIB—Supports the Junos extensions of traceroute and remote operations. Items in this MIB are created when entries are created in the traceRouteCtlTable of the Traceroute MIB. Each item is indexed exactly the same way as it is in the Traceroute MIB. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-traceroute.txt .

  For more information, see *Traceroute MIB*.

- Utility MIB—Provides SNMP support for exposing Junos data and has tables that contain information on each type of data, such as integer and string. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-util.txt .

  For more information, see *Utility MIB*.

- VPN Certificate Objects MIB—Provides support for monitoring the local and CA certificates loaded on the router. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-cert.txt .

  For more information, see *VPN Certificate Objects MIB*.

Related Documentation

- *System Log Monitoring and Troubleshooting Guide for Security Devices*
- *Structure of Management Information MIB*

List of SRX1400, SRX3400, and SRX3600 Services Gateways Supported Enterprise-Specific MIBs

Supported Platforms LN Series, SRX1400, SRX3400, SRX3600

Junos OS supports the following enterprise-specific MIBs:

- Structure of Management Information MIB—Contains object identifiers (OIDs) for the security branch of the MIBs used in Junos OS for SRX Series devices product, services and traps. This MIB is currently supported only by Junos OS for SRX Series devices. It also explains how the Juniper Networks enterprise-specific MIBs are structured. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos12.1x46/topics/reference/mibs/mib-jnx-js-smi.txt. For more information, see *Structure of Management Information MIB*.

- AAA Objects MIB—Provides support for monitoring user authentication, authorization, and accounting through the RADIUS, LDAP, SecurID, and local authentication servers. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-user-aaa.txt .

  For more information, see *AAA Objects MIB*.

- Access Authentication Objects MIB—Provides support for monitoring firewall authentication, including data about the users trying to access firewall-protected resources and the firewall authentication service itself. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-auth.txt .

  For more information, see *Access Authentication Objects MIB*.

- Alarm MIB—Provides support for alarms from the router. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis-alarm.txt .

  For more information, see *Alarm MIB*.

- ATM CoS MIB—Provides support for monitoring Asynchronous Transfer Mode, version 2 (ATM2) virtual circuit (VC) class-of-service (CoS) configurations. It also provides CoS queue statistics for all VCs that have CoS configured. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-atm-cos.txt .

  For more information, see *ATM Class-of-Service MIB*.

- BGP4 V2 MIB—Contains objects used to monitor BGP peer-received prefix counters. It is based upon similar objects in the MIB documented in Internet draft draft-ietf-idr-bgp4-mibv2-03.txt, Definitions of Managed Objects for the Fourth Version of BGP (BGP-4), Second Version. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-bgpmib2.txt .

  For more information, see *BGP4 V2 MIB*.

- BFD MIB—Provides support for monitoring Bidirectional Forwarding Detection (BFD) sessions. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-bfd.txt .

  For more information, see *Bidirectional Forwarding Detection MIB*.

- Chassis MIB—Provides support for environmental monitoring (power supply state, board voltages, fans, temperatures, and air flow) and inventory support for the chassis, System Control Board (SCB), System and Switching Board (SSB), Switching and Forwarding Model (SFM), Flexible PIC Concentrators (FPCs), and PICs. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis.txt .

  For more information, see *Chassis MIBs*.

- Chassis Cluster MIB—Provides information about objects that are used whenever the state of the control link interfaces or fabric link interfaces changes (up to down or down to up) in a chassis cluster deployment. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-jsrpd.txt.

  For more information, see *Chassis Cluster MIB* .

- Configuration Management MIB—Provides notification for configuration changes as SNMP traps. Each trap contains the time at which the configuration change was committed, the name of the user who made the change, and the method by which the change was made. A history of the last 32 configuration changes is kept in **jnxCmChgEventTable**. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-cfgmgmt.txt .

  For more information, see *Configuration Management MIB*.

- Destination Class Usage MIB—Provides support for monitoring packet counts based on the ingress and egress points for traffic transiting your networks. Ingress points are identified by input interface. Egress points are identified by destination prefixes grouped into one or more sets, known as destination classes. One counter is managed per interface per destination class, up to a maximum of 16 counters per interface. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-dcu.txt .

  For more information, see *Destination Class Usage MIB*.

- DNS Objects MIB—Provides support for monitoring DNS proxy queries, requests, responses, and failures. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-dns.txt .

  For more information, see *DNS Objects MIB*.

- Ethernet MAC MIB—Monitors media access control (MAC) statistics on Gigabit Ethernet intelligent queuing (IQ) interfaces. It collects MAC statistics; for example, **inoctets**, **inframes**, **outoctets**, and **outframes** on each source MAC address and virtual LAN (VLAN) ID for each Ethernet port. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-mac.txt .

  For more information, see *Ethernet MAC MIB*.

- Event MIB—Defines a generic trap that can be generated using an op script or event policy. This MIB provides the ability to specify a system log string and raise a trap if that system log string is found. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-event.txt .

  For more information, see *Event MIB*.

- Firewall MIB—Provides support for monitoring firewall filter counters. Routers must have the Internet Processor II ASIC to perform firewall monitoring. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-firewall.txt .

  For more information, see *Firewall MIB*.

- Host Resources MIB—Extends the **hrStorageTable** object, providing a measure of the usage of each file system on the router in percentage. Previously, the objects in the

hrStorageTable measured the usage in allocation units—**hrStorageUsed** and **hrStorageAllocationUnits**—only. Using the percentage measurement, you can more easily monitor and apply thresholds on usage. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-hostresources.txt .

For more information, see *Host Resources MIB*.

- Interface MIB—Extends the standard ifTable (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-if-extensions.txt .

  For more information, see *Interface MIB*.

- IP Forward MIB—Extends the standard IP Forwarding Table MIB (RFC 2096) to include CIDR forwarding information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipforward.txt .

  For more information, see *IP Forward MIB*.

- IPsec Monitoring MIB—Provides operational and statistical information related to the IPsec and IKE tunnels on Juniper Networks routers. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-monitor-asp.txt .

  For more information, see *IPsec Monitoring MIB*.

- IPsec Generic Flow Monitoring Object MIB—Based on **jnx-ipsec-monitor-mib**, this MIB provides support for monitoring IPsec and IPsec VPN management objects. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-flow-mon.txt .

  For more information, see *IPsec Generic Flow Monitoring Object MIB*.

- IPv4 MIB—Provides additional Internet Protocol version 4 (IPv4) address information, supporting the assignment of identical IPv4 addresses to separate interfaces. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipv4.txt .

  For more information, see *IPv4 MIB*.

- License MIB—Extends SNMP support to licensing information, and introduces SNMP traps that alert users when the licenses are about to expire, expire, or when the total number of users exceeds the number specified in the license. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-license.txt .

  For more information, see *License MIB*.

- Logical Systems MIB—Provides support for logical systems security profile. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-lsys-securityprofile.txt.

For more information, see *Logical Systems MIB*.

- NAT Objects MIB—Provides support for monitoring network address translation (NAT). This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-nat.txt .

  For more information, see *NAT Objects MIB*.

- Packet Forwarding Engine MIB—Provides notification statistics for Packet Forwarding Engines. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-pfe.txt .

  For more information, see *Packet Forwarding Engine MIB*.

- Ping MIB—Extends the standard Ping MIB control table (RFC 2925). Items in this MIB are created when entries are created in **pingCtlTable** of the Ping MIB. Each item is indexed exactly as it is in the Ping MIB. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ping.txt .

  For more information, see *PING MIB*.

- Policy Objects MIB—Provides support for monitoring the security policies that control the flow of traffic from one zone to another. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-policy.txt .

  For more information, see *Policy Objects MIB*.

- Reverse-Path-Forwarding MIB—Monitors statistics for traffic that is rejected because of reverse-path-forwarding (RPF) processing. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-rpf.txt .

  NOTE: **The enterprise-specific RPF MIB is not supported on EX Series Ethernet Switches.**

  For more information, see *Reverse Path Forwarding MIB*.

- RMON Events and Alarms MIB—Supports the Junos OS extensions to the standard Remote Monitoring (RMON) Events and Alarms MIB (RFC 2819). The extension augments **alarmTable** with additional information about each alarm. Two new traps are also defined to indicate when problems are encountered with an alarm. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-rmon.txt .

  For more information, see *RMON Events and Alarms MIB*.

- Security Interface Extension Objects MIB—Provides support for the security management of interfaces. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-if-ext.txt .

  For more information, see *Security Interface Extension Objects MIB*.

- Security Screening Objects MIB—Defines the MIB for the Juniper Networks Enterprise Firewall screen functionality. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-screening.txt .

  For more information, see *Security Screening Objects MIB*.

- Source Class Usage MIB—Counts packets sent to customers by performing a lookup on the IP source address and the IP destination address. The Source Class Usage (SCU) MIB makes it possible to track traffic originating from specific prefixes on the provider core and destined for specific prefixes on the customer edge. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-scu.txt .

  For more information, see *Source Class Usage MIB*.

- SPU Monitoring MIB—Provides support for monitoring SPUs on SRX5600 and SRX5800 devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-spu-monitoring.txt .

  For more information, see *SPU Monitoring Objects MIB*.

- System Log MIB—Enables notification of an SNMP trap-based application when an important system log message occurs. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-syslog.txt .

  For more information, see *System Log MIB*.

- Traceroute MIB—Supports the Junos OS extensions of traceroute and remote operations. Items in this MIB are created when entries are created in the **traceRouteCtlTable** of the Traceroute MIB. Each item is indexed exactly the same way as it is in the Traceroute MIB. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-traceroute.txt .

  For more information, see *Traceroute MIB*.

- Utility MIB—Provides SNMP support for exposing Junos OS data and has tables that contain information about each type of data, such as integer and string. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-util.txt .

For more information, see *Utility MIB*.

- VPN Certificate Objects MIB—Provides support for monitoring the local and CA certificates loaded on the router. This MIB is currently supported by Junos OS for SRX Series devices only. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-cert.txt .

  For more information, see *VPN Certificate Objects MIB*.

Related Documentation

- *Structure of Management Information MIB*

List of SRX5400, SRX5600 and SRX5800 Services Gateways Supported Enterprise-Specific MIBs

Supported Platforms LN Series, SRX5400, SRX5600, SRX5800

Junos OS supports the following enterprise-specific MIBs:

- Structure of Management Information MIB—Contains object identifiers (OIDs) for the security branch of the MIBs used in Junos OS for SRX Series devices product, services and traps. This MIB is currently supported only by Junos OS for SRX Series devices. It also explains how the Juniper Networks enterprise-specific MIBs are structured. For a downloadable version of this MIB, http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-smi.txt . For more information, see *Structure of Management Information MIB*.

- AAA Objects MIB—Provides support for monitoring user authentication, authorization, and accounting through the RADIUS, LDAP, SecurID, and local authentication servers. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-user-aaa.txt .

  For more information, see *AAA Objects MIB*.

- Access Authentication Objects MIB—Provides support for monitoring firewall authentication, including data about the users trying to access firewall-protected resources and the firewall authentication service itself. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-auth.txt .

  For more information, see *Access Authentication Objects MIB*.

- Alarm MIB—Provides support for alarms from the router. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis-alarm.txt .

  For more information, see *Alarm MIB*.

- ATM CoS MIB—Provides support for monitoring Asynchronous Transfer Mode, version 2 (ATM2) virtual circuit (VC) class-of-service (CoS) configurations. It also provides

CoS queue statistics for all VCs that have CoS configured. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-atm-cos.txt .

For more information, see *ATM Class-of-Service MIB*.

- BGP4 V2 MIB—Contains objects used to monitor BGP peer-received prefix counters. It is based upon similar objects in the MIB documented in Internet draft draft-ietf-idr-bgp4-mibv2-03.txt, Definitions of Managed Objects for the Fourth Version of BGP (BGP-4), Second Version. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-bgpmib2.txt .

  For more information, see *BGP4 V2 MIB*.

- BFD MIB—Provides support for monitoring Bidirectional Forwarding Detection (BFD) sessions. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-bfd.txt . .

  For more information, see *Bidirectional Forwarding Detection MIB*.

- Chassis MIB—Provides support for environmental monitoring (power supply state, board voltages, fans, temperatures, and air flow) and inventory support for the chassis, System Control Board (SCB), System and Switching Board (SSB), Switching and Forwarding Model (SFM), Flexible PIC Concentrators (FPCs), and PICs. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-chassis.txt .

  For more information, see *Chassis MIBs*.

- Chassis Cluster MIB—Provides information about objects that are used whenever the state of the control link interfaces or fabric link interfaces changes (up to down or down to up) in a chassis cluster deployment. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-jsrpd.txt.

  For more information, see *Chassis Cluster MIB* .

- Configuration Management MIB—Provides notification for configuration changes as SNMP traps. Each trap contains the time at which the configuration change was committed, the name of the user who made the change, and the method by which the change was made. A history of the last 32 configuration changes is kept in **jnxCmChgEventTable**. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-cfgmgmt.txt .

  For more information, see *Configuration Management MIB*.

- Destination Class Usage MIB—Provides support for monitoring packet counts based on the ingress and egress points for traffic transiting your networks. Ingress points are identified by input interface. Egress points are identified by destination prefixes grouped into one or more sets, known as destination classes. One counter is managed per interface per destination class, up to a maximum of 16 counters per interface. For a downloadable version of this MIB, see

http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-dcu.txt .

For more information, see *Destination Class Usage MIB*.

- DNS Objects MIB—Provides support for monitoring DNS proxy queries, requests, responses, and failures. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-dns.txt .

  For more information, see *DNS Objects MIB*.

- Ethernet MAC MIB—Monitors media access control (MAC) statistics on Gigabit Ethernet intelligent queuing (IQ) interfaces. It collects MAC statistics; for example, **inoctets**, **inframes**, **outoctets**, and **outframes** on each source MAC address and virtual LAN (VLAN) ID for each Ethernet port. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/jnx-mac.txt .

  For more information, see *Ethernet MAC MIB*.

- Event MIB—Defines a generic trap that can be generated using an op script or event policy. This MIB provides the ability to specify a system log string and raise a trap if that system log string is found. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-event.txt .

  For more information, see *Event MIB*.

- Firewall MIB—Provides support for monitoring firewall filter counters. Routers must have the Internet Processor II ASIC to perform firewall monitoring. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-firewall.txt .

  For more information, see *Firewall MIB*.

- Host Resources MIB—Extends the **hrStorageTable** object, providing a measure of the usage of each file system on the router in percentage. Previously, the objects in the **hrStorageTable** measured the usage in allocation units—**hrStorageUsed** and **hrStorageAllocationUnits**—only. Using the percentage measurement, you can more easily monitor and apply thresholds on usage. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-hostresources.txt .

  For more information, see *Host Resources MIB*.

- Interface MIB—Extends the standard ifTable (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-if-extensions.txt .

  For more information, see *Interface MIB*.

- IP Forward MIB—Extends the standard IP Forwarding Table MIB (RFC 2096) to include CIDR forwarding information. For a downloadable version of this MIB, see

http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipforward.txt .

For more information, see *IP Forward MIB*.

- IPsec Generic Flow Monitoring Object MIB—Based on **jnx-ipsec-monitor-mib**, this MIB provides support for monitoring IPsec and IPsec VPN management objects. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-flow-mon.txt .

  For more information, see *IPsec Generic Flow Monitoring Object MIB*.

- IPsec Monitoring MIB—Provides operational and statistical information related to the IPsec and IKE tunnels on Juniper Networks routers. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipsec-monitor-asp.txt .

  For more information, see *IPsec Monitoring MIB*.

- IPv4 MIB—Provides additional Internet Protocol version 4 (IPv4) address information, supporting the assignment of identical IPv4 addresses to separate interfaces. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ipv4.txt .

  For more information, see *IPv4 MIB*.

- License MIB—Extends SNMP support to licensing information, and introduces SNMP traps that alert users when the licenses are about to expire, expire, or when the total number of users exceeds the number specified in the license. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-license.txt .

  For more information, see *License MIB*.

- Logical Systems MIB—Provides support for logical systems security profile. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-lsys-securityprofile.txt.

  For more information, see *Logical Systems MIB*.

- Network Address Translation (NAT) Objects MIB—Provides support for monitoring network address translation (NAT). This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-nat.txt . .

  For more information, see *NAT Objects MIB*.

- Packet Forwarding Engine MIB—Provides notification statistics for Packet Forwarding Engines. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-pfe.txt .

  For more information, see *Packet Forwarding Engine MIB*.

- Ping MIB—Extends the standard Ping MIB control table (RFC 2925). Items in this MIB are created when entries are created in pingCtlTable of the Ping MIB. Each item is indexed exactly as it is in the Ping MIB. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-ping.txt .

  For more information, see *PING MIB*.

- Policy Objects MIB—Provides support for monitoring the security policies that control the flow of traffic from one zone to another. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-policy.txt .

  For more information, see *Policy Objects MIB*.

- Reverse-Path-Forwarding MIB—Monitors statistics for traffic that is rejected because of reverse-path-forwarding (RPF) processing. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-rpf.txt .

  NOTE: The enterprise-specific RPF MIB is not supported on EX Series Ethernet Switches.

  For more information, see *Reverse Path Forwarding MIB*.

- RMON Events and Alarms MIB—Supports the Junos OS extensions to the standard Remote Monitoring (RMON) Events and Alarms MIB (RFC 2819). The extension augments alarmTable with additional information about each alarm. Two new traps are also defined to indicate when problems are encountered with an alarm. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-rmon.txt .

  For more information, see *RMON Events and Alarms MIB*.

- Security Interface Extension Objects MIB—Provides support for the security management of interfaces. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-if-ext.txt .

  For more information, see *Security Interface Extension Objects MIB*.

- Security Screening Objects MIB—Defines the MIB for the Juniper Networks Enterprise Firewall screen functionality. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-screening.txt .

  For more information, see *Security Screening Objects MIB*.

- Source Class Usage MIB—Counts packets sent to customers by performing a lookup on the IP source address and the IP destination address. The Source Class Usage (SCU) MIB makes it possible to track traffic originating from specific prefixes on the provider core and destined for specific prefixes on the customer edge. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-scu.txt .

  For more information, see *Source Class Usage MIB*.

- SPU Monitoring MIB—Provides support for monitoring SPUs on SRX5600 and SRX5800 devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-spu-monitoring.txt .

  For more information, see *SPU Monitoring Objects MIB*.

- System Log MIB—Enables notification of an SNMP trap-based application when an important system log message occurs. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-syslog.txt .

  For more information, see *System Log MIB*.

- Traceroute MIB—Supports the Junos OS extensions of traceroute and remote operations. Items in this MIB are created when entries are created in the traceRouteCtlTable of the Traceroute MIB. Each item is indexed exactly the same way as it is in the Traceroute MIB. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-traceroute.txt .

  For more information, see *Traceroute MIB*.

- Utility MIB—Provides SNMP support for exposing Junos OS data and has tables that contain information on each type of data, such as integer and string. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-util.txt .

  For more information, see *Utility MIB*.

- VPN Certificate Objects MIB—Provides support for monitoring the local and CA certificates loaded on the router. This MIB is currently supported only by Junos OS for SRX Series devices. For a downloadable version of this MIB, see http://www.juniper.net/techpubs/en_US/junos15.1x49/topics/reference/mibs/mib-jnx-js-cert.txt .

  For more information, see *VPN Certificate Objects MIB*.

Related Documentation

- *Structure of Management Information MIB*

Enterprise-Specific MIBs and Supported Devices

Supported Platforms ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, T Series

Table 5 on page 54 lists the enterprise-specific MIBs that are supported on various devices running the Junos OS.

NOTE: In this table, a value of 1 in any of the platform columns (M, MX, T, EX, J, and SRX) denotes that the corresponding MIB is supported on that particular platform. A value of 0 denotes that the MIB is not supported on the platform.

NOTE: This topic uses the following classification for SRX Series devices: Low-End (SRX100, SRX110, SRX210, SRX220, and SRX240), Mid-Range (SRX550 and SRX650), and High-End (SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800).

Table 5: Enterprise-Specific MIBs and Supported Devices

Enterprise-Specific MIB	Platforms									
								SRX		
	ACX	M	T	J	MX	EX	PTX	Low-End	Mid-Range	High-End
AAA Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-user-aaa.txt	0	1	1	0	0	0	0	1	1	
Access Authentication Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-auth.txt	0	0	0	0	1	0	1	1	1	
Alarm MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-chassis-alarm.txt	1	1	1	1	1	1	1	1	1	
Analyzer MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-analyzer.txt	0	0	0	1	0	0	0	0	0	
Antivirus Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-utm-av.txt	0	0	0	0	0	0	1	0	0	
ATM Class-of-Service MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-atm-cos.txt	1	1	1	0	0	0	1	0	1	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
	ACX	M	T	J	MX	EX	PTX	SRX		
								Low-End	Mid-Range	High-End
ATM MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-atm.txt	1	1	1	0	0	0	0	0	0	
BGP4 V2 MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-bgpmib2.txt	1	1	1	1	1	1	1	1	1	
Bidirectional Forwarding Detection MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-bfd.txt	1	1	1	1	1	1	1	1	1	
Chassis Forwarding MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-chassis-fwdd.txt	1	0	0	0	0	1	1	0	0	
Chassis MIBs http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-chassis.txt http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-chas-defines.txt	1	1	1	1	1	1	1	1	1	
Chassis Cluster MIBs http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-jsrpd.txt	0	0	0	0	0	0	0	1	1	
Class-of-Service MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-cos.txt	1	1	1	1	1	1	0	0	1	
Configuration Management MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-cfgmgmt.txt	1	1	1	1	1	1	1	1	1	
Destination Class Usage MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-dcu.txt	1	1	1	0	1	0	0	1	1	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
	ACX	M	T	J	MX	EX	PTX	SRX		
								Low-End	Mid-Range	High-End
DHCP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-jdhcp.txt	1	1	1	0	0	0	0	0	0	
DHCPv6 MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-jdhcpv6.txt	0	1	1	1	0	0	0	0	0	
Digital Optical Monitoring MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-dom.txt	1	0	1	0	0	0	0	0		
DNS Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-dns.txt	0	0	0	0	0	0	0	1	1	
Dynamic Flow Capture MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-dfc.txt	0	1	1	1	0	0	0	0	0	
Ethernet MAC MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/jnx-mac.txt	0	1	1	1	1	1	0	0	1	
Event MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-event.txt	1	1	1	1	1	1	1	1	1	
EX Series MAC Notification MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ex-mac-notification.txt	0	0	0	1	0	0	0	0	0	
EX Series SMI MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ex-smi.txt	0	0	0	1	0	0	0	0	0	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
								SRX		
	ACX	M	T	J	MX	EX	PTX	Low-End	Mid-Range	High-End
Experimental MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-exp.txt	1	1	1	1	1	0	0	0	0	
Firewall MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-firewall.txt	1	1	1	1	1	1	1	1	1	
Flow Collection Services MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-coll.txt	1	1	1	0	0	0	0	0	0	
Host Resources MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-hostresources.txt	1	1	1	1	1	0	1	1	1	
Interface MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-if-extensions.txt	1	1	1	1	1	1	1	1	1	
IP Forward MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ipforward.txt	1	1	1	1	1	1	1	1	1	
IPsec Generic Flow Monitoring Object MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ipsec-flow-mon.txt	0	0	0	0	1	0	0	1	1	1
IPsec Monitoring MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ipsec-monitor-asp.txt	1	1	1	0	1	0	0	1	0	
IPsec VPN Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-ipsec-vpn.txt	0	0	0	0	0	0	1	1	0	0

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
								SRX		
	ACX	M	T	J	MX	EX	PTX	Low-End	Mid-Range	High-End
IPv4 MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ipv4.txt	1	1	1	1	1	1	1	1	1	
IPv6 and ICMPv6 MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ipv6.txt	1	1	1	1	0	1	1	1		
L2ALD MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-l2ald.txt	0	0	1	1	0	1	0	0	0	
L2CP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-l2cp-features.txt	0	0	0	1	0	0	0			
L2TP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-l2tp.txt	0	1	1	0	0	0	0	0	0	
LDP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ldp.txt	1	1	1	0	0	1	0	0	1	
License MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-license.txt	1	1	1	0	0	0	1	1	1	
Logical Systems MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-lsys-securityprofile.txt	0	0	0	0	0	0	0	1	1	
MIMSTP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-mimstp.txt	0	0	1	0	1	0	0	0	0	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
	ACX	M	T	J	MX	EX	PTX	SRX		
								Low-End	Mid-Range	High-End
MPLS LDP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-mpls-ldp.txt	1	1	1	1	0	1	0	0	0	
MPLS MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-mpls.txt	1	1	1	1	1	1	0	0	1	
MVPN MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-mvpn.txt and http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-l2vpn-mvpn.txt	1	1	1	1	1	1	1	1	1	
NAT Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-nat.txt	0	0	0	0	1	0	1	1	1	
NAT Resources-Monitoring MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/ mibs/mib-jnx-sp-nat.txt	1	1	1	0	0	0	0	0		
OTN Interface Management MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-otn.txt	1	1	1	0	0	0	0	0	0	
Packet Forwarding Engine MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-pfe.txt	1	1	1	0	1	1	1	1	1	
Packet Mirror MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-packet-mirror.txt	0	1	0	1	0	0	0	0	0	
PAE Extension MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-pae-extension.txt	0	0	0	1	0	0	0	0	0	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
								SRX		
	ACX	M	T	J	MX	EX	PTX	Low-End	Mid-Range	High-End
Passive Monitoring MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-pmon.txt	0	1	1	1	0	0	0	0	0	
Ping MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ping.txt	1	1	1	1	1	0	1	1	1	
Policy Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-policy.txt	0	0	0	0	1	0	1	1	1	
Power Supply Unit MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-power-supply-unit.txt	0	0	0	1	0	1	0	0	0	
PPP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ppp.txt.	0	1	1	0	0	0	0	0	0	
PPPoE MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-pppoe.txt.	0	1	1	0	0	0	0	0	0	
Pseudowire ATM MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-pwatm.txt	0	1	0	1	0	0	0	0	0	
Psuedowire TDM MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-pwtdm.txt .	1	1	1	0	0	0	0	0	0	
PTP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-ptp-mgmt-defs.txt	0	0	0	1	0	0	0	0	0	
Real-Time Performance Monitoring MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-rpm.txt	1	1	1	1	1	0	1	0	0	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
	ACX	M	T	J	MX	EX	PTX	SRX		
								Low-End	Mid-Range	High-End
Reverse-Path-Forwarding MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-rpf.txt	1	1	1	0	1	1	1	1		
RMON Events and Alarms MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-rmon.txt	1	1	1	1	0	1	1	1	1	
RSVP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-rsvp.txt	1	1	1	1	0	0	0	0	0	
Security Interface Extension Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-if-ext.txt	0	0	0	0	1	0	1	1	1	
Security Screening Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-screening.txt	0	0	0	0	0	0	0	0	1	
Services PIC MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-sp.txt	1	1	1	0	0	0	0	0	0	
SNMP IDP MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-idp.txt.	0	0	0	0	0	1	1	1	0	
SONET APS MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-sonetaps.txt	0	1	1	1	0	0	0	0	0	
SONET/SDH Interface Management MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-sonet.txt	1	1	1	0	0	0	0	0	0	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
								SRX		
	ACX	M	T	J	MX	EX	PTX	Low-End	Mid-Range	High-End
Source Class Usage MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-scu.txt	1	1	1	0	0	0	0	0	1	
SPU Monitoring MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-spu-monitoring.txt	0	0	0	0	0	0	1	1	1	
Structure of Management Information MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-smi.txt	1	1	1	1	1	0	1	1	1	
Subscriber MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-subscriber.txt	1	0	1	0	0	0	0	0	0	
System Log MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-syslog.txt	0	1	1	1	1	1	1	1	1	
Traceroute MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-traceroute.txt	0	1	1	1	1	1	1	1	1	
Utility MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-util.txt	0	1	1	1	1	1	1	1	1	
Virtual Chassis MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-virtualchassis.txt	0	0	0	1	1	0	0	0	0	
VLAN MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-vlan.txt	0	0	0	1	0	0	0	0	0	

Table 5: Enterprise-Specific MIBs and Supported Devices *(continued)*

Enterprise-Specific MIB	Platforms									
								SRX		
	ACX	M	T	J	MX	EX	PTX	Low-End	Mid-Range	High-End
VPLS MIBs • http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-vpls-generic.txt • http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-vpls-ldp.txt • http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-vpls-bgp.txt	1	1	1	1	0	0	0	0	0	
VPN Certificate Objects MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-js-cert.txt	0	0	0	0	1	0	1	1	1	
VPN MIB http://www.juniper.net/techpubs/en_US/junos12.1/topics/reference/mibs/mib-jnx-vpn.txt	1	1	1	0	1	0	0	0	0	

Related Documentation

- Juniper Networks Enterprise-Specific MIBs on page 32
- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Standard SNMP MIBs Supported by Junos OS on page 13
- Loading MIB Files to a Network Management System on page 111

MIB Support Details

Supported Platforms LN Series, M Series, MX Series, SRX Series, T Series

Table 6 on page 63 shows enterprise-specific MIB objects supported by Junos OS and provides notes detailing how they are handled when a routing instance is specified in an SNMP request. An en dash (–) indicates that the item is not applicable.

Table 6: MIB Support for Routing Instances (Juniper Networks MIBs)

Object	Support Class	Description/Notes
jnxProducts(1)	–	Product Object IDs
jnxServices(2)	–	Services

Table 6: MIB Support for Routing Instances (Juniper Networks MIBs) *(continued)*

Object	Support Class	Description/Notes
jnxMibs(3) **jnxBoxAnatomy(1)**	Class 3	Objects are exposed only for the default logical system.
mpls(2)	Class 2	All instances within a logical system are exposed. Data will not be segregated down to the routing instance level.
ifJnx(3)	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxAlarms(4)	Class 3	Objects are exposed only for the default logical system.
jnxFirewalls(5)	Class 4	Data is not segregated by routing instance. All instances are exposed.
jnxDCUs(6)	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxPingMIB(7)	Class 3	Objects are exposed only for the default logical system.
jnxTraceRouteMIB(8)	Class 3	Objects are exposed only for the default logical system.
jnxATM(10)	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxIpv6(11)	Class 4	Data is not segregated by routing instance. All instances are exposed.
jnxIpv4(12)	Class 1	**jnxIpv4AddrTable(1)**. Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxRmon(13)	Class 3	**jnxRmonAlarmTable(1)**. Objects are exposed only for the default logical system.
jnxLdp(14)	Class 2	**jnxLdpTrapVars(1)**. All instances within a logical system are exposed. Data will not be segregated down to the routing instance level.

Table 6: MIB Support for Routing Instances (Juniper Networks MIBs) *(continued)*

Object	Support Class	Description/Notes
jnxCos(15) **jnxCosIfqStatsTable(1)** **jnxCosFcTable(2)** **jnxCosFcIdTable(3)** **jnxCosQstatTable(4)**	Class 3	Objects are exposed only for the default logical system.
jnxScu(16) **jnxScuStatsTable(1)**	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxRpf(17) **jnxRpfStatsTable(1)**	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxCfgMgmt(18)	Class 3	Objects are exposed only for the default logical system.
jnxPMon(19) **jnxPMonFlowTable(1)** **jnxPMonErrorTable(2)** **jnxPMonMemoryTable(3)**	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxSonet(20) **jnxSonetAlarmTable(1)**	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxAtmCos(21) **jnxCosAtmVcTable(1)** **jnxCosAtmScTable(2)** **jnxCosAtmVcQstatsTable(3)** **jnxCosAtmTrunkTable(4)**	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
ipSecFlowMonitorMIB(22)	–	–
jnxMac(23) **jnxMacStats(1)**	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
apsMIB(24)	Class 3	Objects are exposed only for the default logical system.
jnxChassisDefines(25)	Class 3	Objects are exposed only for the default logical system.

Table 6: MIB Support for Routing Instances (Juniper Networks MIBs) *(continued)*

Object	Support Class	Description/Notes
jnxVpnMIB(26)	Class 2	All instances within a logical system are exposed. Data will not be segregated down to the routing instance level.
jnxSericesInfoMib(27)	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxCollectorMIB(28)	Class 1	Only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.
jnxHistory(29)	–	–
jnxSpMIB(32)	Class 3	Objects are exposed only for the default logical system.

Table 7 on page 67 shows Class 1 MIB objects (standard and enterprise-specific MIBs) supported by Junos OS. With Class 1 objects, only those logical interfaces (and their parent physical interfaces) that belong to a specific routing instance are exposed.

Table 7: Class 1 MIB Objects (Standard and Juniper MIBs)

Class	MIB	Objects
Class 1	**802.3ad.mib**	(**dot3adAgg**) MIB objects: **dot3adAggTable** **dot3adAggPortListTable** (**dot3adAggPort**) **dot3adAggPortTable** **dot3adAggPortStatsTable** **dot3adAggPortDebugTable**
	rfc2863a.mib	**ifTable** **ifXTable** **ifStackTable**
	rfc2011a.mib	**ipAddrTable** **ipNetToMediaTable**
	rtmib.mib	**ipForward** (**ipCidrRouteTable**)
	rfc2665a.mib	**dot3StatsTable** **dot3ControlTable** **dot3PauseTable**
	rfc2495a.mib	**dsx1ConfigTable** **dsx1CurrentTable** **dsx1IntervalTable** **dsx1TotalTable** **dsx1FarEndCurrentTable** **dsx1FarEndIntervalTable** **dsx1FarEndTotalTable** **dsx1FracTable** ...
	rfc2496a.mib	**dsx3 (dsx3ConfigTable)**
	rfc2115a.mib	**frDlcmiTable** (and related MIB objects)
	rfc3592.mib	**sonetMediumTable** (and related MIB objects)

Table 7: Class 1 MIB Objects (Standard and Juniper MIBs) *(continued)*

Class	MIB	Objects
	rfc3020.mib	**mfrMIB** **mfrBundleTable** **mfrMibBundleLinkObjects** **mfrBundleIfIndexMappingTable** (and related MIB objects)
	ospf2mib.mib	All objects
	ospf2trap.mib	All objects
	bgpmib.mib	All objects
	rfc2819a.mib	Example: **etherStatsTable**

Table 7: Class 1 MIB Objects (Standard and Juniper MIBs) *(continued)*

Class	MIB	Objects
Class 1	**rfc2863a.mib**	Examples: **ifXtable** **ifStackTable**
	rfc2665a.mib	**etherMIB**
	rfc2515a.mib	**atmMIB** objects Examples: **atmInterfaceConfTable** **atmVplTable** **atmVclTable**
	rfc2465.mib	**ip-v6mib** Examples: **ipv6IfTable** **ipv6AddrPrefixTable** **ipv6NetToMediaTable** **ipv6RouteTable**
	rfc2787a.mib	**vrrp mib**
	rfc2932.mib	**ipMRouteMIB** **ipMRouteStdMIB**
	mroutemib.mib	**ipMRoute1MIBObjects**
	isismib.mib	**isisMIB**
	pimmib.mib	**pimMIB**
	msdpmib.mib	**msdpmib**
	jnx-if-extensions.mib	Examples: **ifJnxTable** **ifChassisTable**
	jnx-dcu.mib	**jnxDCUs**
	jnx-atm.mib	

Table 7: Class 1 MIB Objects (Standard and Juniper MIBs) *(continued)*

Class	MIB	Objects
		Examples: **jnxAtmIfTable** **jnxAtmVCTable** **jnxAtmVpTable**
	jnx-ipv4.mib	**jnxipv4** Example: **jnxIpv4AddrTable**
	jnx-cos.mib	Examples: **jnxCosIfqStatsTable** **jnxCosQstatTable**
	jnx-scu.mib	Example: **jnxScuStatsTable**
	jnx-rpf.mib	Example: **jnxRpfStatsTable**
	jnx-pmon.mib	Example: **jnxPMonFlowTable**
	jnx-sonet.mib	Example: **jnxSonetAlarmTable**
Class 1	**jnx-atm-cos.mib**	Examples: **jnxCosAtmVcTable** **jnxCosAtmVcScTable** **jnxCosAtmVcQstatsTable** **jnxCosAtmTrunkTable**
	jnx-mac.mib	Example: **jnxMacStatsTable**
	jnx-services.mib	Example: **jnxSvcFlowTableAggStatsTable**
	jnx-coll.mib	**jnxCollectorMIB** Examples: **jnxCollPicIfTable** **jnxCollFileEntry**

Table 8 on page 71 shows Class 2 MIB objects (standard and enterprise-specific MIBs) supported by Junos OS. With Class 2 objects, all instances within a logical system are exposed. Data will not be segregated down to the routing instance level.

Table 8: Class 2 MIB Objects (Standard and Juniper MIBs)

Class	MIB	Objects
Class 2	**rfc3813.mib**	**mplsLsrStdMIB** Examples: **mplsInterfaceTable** **mplsInSegmentTable** **mplsOutSegmentTable** **mplsLabelStackTable** **mplsXCTable** (and related MIB objects)
	igmpmib.mib	**igmpStdMIB**
	l3vpnmib.mib	**mplsVpnmib**
	jnx-mpls.mib	Example: **mplsLspList**
	jnx-ldp.mib	**jnxLdp** Example: **jnxLdpStatsTable**
	jnx-vpn.mib	**jnxVpnMIB**
	jnx-bgpmib2.mib	**jnxBgpM2Experiment**

Table 9 on page 72 shows Class 3 MIB objects (standard and enterprise-specific MIBs) supported by Junos OS. With Class 3, objects are exposed only for the default logical system.

Table 9: Class 3 MIB Objects (Standard and Juniper MIBs)

Class	MIB	Objects
Class 3	rfc2819a.mib	rmonEvents alarmTable logTable eventTable agentxMIB
	rfc2925a.mib	pingmib
	rfc2925b.mib	tracerouteMIB
	jnxchassis.mib	jnxBoxAnatomy
	jnx-chassis-alarm.mib	jnxAlarms
	jnx-ping.mib	jnxPingMIB
	jnx-traceroute.mib	jnxTraceRouteMIB
	jnx-rmon.mib	jnxRmonAlarmTable
	jnx-cos.mib	Example: jnxCosFcTable
	jnx-cfgmgmt.mib	Example: jnxCfgMgmt
	jnx-sonetaps.mib	apsMIBObjects
	jnx-sp.mib	jnxSpMIB
	ggsn.mib	ejnmobileipABmib
	rfc1907.mib	snmpModules
	snmpModules	Examples: snmpMIB snmpFrameworkMIB

Table 10 on page 73 shows Class 4 MIB objects (standard and enterprise-specific MIBs) supported by Junos OS. With Class 4 objects, data is not segregated by routing instance. All instances are exposed.

Table 10: Class 4 MIB Objects (Standard and Juniper MIBs)

Class	MIB	Objects
Class 4	system	Example: sysORTable
	rfc2011a.mib	ip (ipDefaultTTL, ipInReceives) icmp
	rfc2012a.mib	tcp tcpConnTable ipv6TcpConnTable
	rfc2013a.mib	udp udpTable ipv6UdpTable
	rfc2790a.mib	hrSystem
	rfc2287a.mib	sysApplOBJ
	jnx-firewall.mib	jnxFirewalls
	jnx-ipv6.mib	jnxIpv6

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- Trap Support for Routing Instances on page 178

SNMP MIB Objects Supported by Junos OS for the Set Operation

Supported Platforms LN Series, SRX Series

Table 11 on page 73 lists the SNMP MIB objects that are supported by Junos OS for the **snmp set** operation.

Table 11: SNMP MIB Objects

Object Name	Object Identifier
RFC 1907	
sysContact	1.3.6.1.2.1.1.4
sysName	1.3.6.1.2.1.1.5
sysLocation	1.3.6.1.2.1.1.6

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
snmpEnableAuthenTraps	1.3.6.1.2.1.11.30
RFC 2819a	
alarmInterval	1.3.6.1.2.1.16.3.1.1.2
alarmVariable	1.3.6.1.2.1.16.3.1.1.2
alarmSampleType	1.3.6.1.2.1.16.3.1.1.4
alarmStartupAlarm	1.3.6.1.2.1.16.3.1.1.6
alarmRisingThreshold	1.3.6.1.2.1.16.3.1.1.7
alarmFallingThreshold	1.3.6.1.2.1.16.3.1.1.8
alarmRisingEventIndex	1.3.6.1.2.1.16.3.1.1.9
alarmFallingEventIndex	1.3.6.1.2.1.16.3.1.1.10
alarmOwner	1.3.6.1.2.1.16.3.1.1.11
alarmStatus	1.3.6.1.2.1.16.3.1.1.12
eventDescription	1.3.6.1.2.1.16.9.1.1.2
eventType	1.3.6.1.2.1.16.9.1.1.3
eventCommunity	1.3.6.1.2.1.16.9.1.1.4
eventOwner	1.3.6.1.2.1.16.9.1.1.6
eventStatus	1.3.6.1.2.1.16.9.1.1.7
RFC 2925a	
pingMaxConcurrentRequests	1.3.6.1.2.1.80.1.1
pingCtlTargetAddressType	1.3.6.1.2.1.80.1.2.1.3
pingCtlTargetAddress	1.3.6.1.2.1.80.1.2.1.4
pingCtlDataSize	1.3.6.1.2.1.80.1.2.1.5
pingCtlTimeOut	1.3.6.1.2.1.80.1.2.1.6
pingCtlProbeCount	1.3.6.1.2.1.80.1.2.1.7

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
pingCtlAdminStatus	1.3.6.1.2.1.80.1.2.1.8
pingCtlDataFill	1.3.6.1.2.1.80.1.2.1.9
pingCtlFrequency	1.3.6.1.2.1.80.1.2.1.10
pingCtlMaxRows	1.3.6.1.2.1.80.1.2.1.11
pingCtlStorageType	1.3.6.1.2.1.80.1.2.1.12
pingCtlTrapGeneration	1.3.6.1.2.1.80.1.2.1.13
pingCtlTrapProbeFailureFilter	1.3.6.1.2.1.80.1.2.1.14
pingCtlTrapTestFailureFilter	1.3.6.1.2.1.80.1.2.1.15
pingCtlType	1.3.6.1.2.1.80.1.2.1.16
pingCtlDescr	1.3.6.1.2.1.80.1.2.1.17
pingCtlSourceAddressType	1.3.6.1.2.1.80.1.2.1.18
pingCtlSourceAddress	1.3.6.1.2.1.80.1.2.1.19
pingCtlIfIndex	1.3.6.1.2.1.80.1.2.1.20
pingCtlByPassRouteTable	1.3.6.1.2.1.80.1.2.1.21
pingCtlDSField	1.3.6.1.2.1.80.1.2.1.22
pingCtlRowStatus	1.3.6.1.2.1.80.1.2.1.23
RFC 2925B	
traceRouteMaxConcurrentRequests	1.3.6,1.2.1.81.1.1
traceRouteCtlTargetAddressType	1.3.6,1.2.1.81.1.2.1.3
traceRouteCtlTargetAddress	1.3.6,1.2.1.81.1.2.1.4
traceRouteCtlByPassRouteTable	1.3.6,1.2.1.81.1.2.1.5
traceRouteCtlDataSize	1.3.6,1.2.1.81.1.2.1.6
traceRouteCtlTimeOut	1.3.6,1.2.1.81.1.2.1.7
traceRouteCtlProbesPerHop	1.3.6,1.2.1.81.1.2.1.8

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
traceRouteCtlPort	1.3.6,1.2.1.81.1.2.1.9
traceRouteCtlMaxTtl	1.3.6,1.2.1.81.1.2.1.10
traceRouteCtlDSField	1.3.6,1.2.1.81.1.2.1.11
traceRouteCtlSourceAddressType	1.3.6,1.2.1.81.1.2.1.12
traceRouteCtlSourceAddress	1.3.6,1.2.1.81.1.2.1.13
traceRouteCtlIfIndex	1.3.6,1.2.1.81.1.2.1.14
traceRouteCtlMiscOptions	1.3.6,1.2.1.81.1.2.1.15
traceRouteCtlMaxFailure	1.3.6,1.2.1.81.1.2.1.16
traceRouteCtlDontFragment	1.3.6,1.2.1.81.1.2.1.17
traceRouteCtlInitialTtl	1.3.6,1.2.1.81.1.2.1.18
traceRouteCtlFrequency	1.3.6,1.2.1.81.1.2.1.19
traceRouteCtlStorageType	1.3.6,1.2.1.81.1.2.1.20
traceRouteCtlAdminStatus	1.3.6,1.2.1.81.1.2.1.21
traceRouteCtlDescr	1.3.6,1.2.1.81.1.2.1.22
traceRouteCtlMaxRows	1.3.6,1.2.1.81.1.2.1.23
traceRouteCtlTrapGeneration	1.3.6,1.2.1.81.1.2.1.24
traceRouteCtlCreateHopEntries	1.3.6,1.2.1.81.1.2.1.25
traceRouteCtlType	1.3.6,1.2.1.81.1.2.1.26
traceRouteCtlRowStatus	1.3.6,1.2.1.81.1.2.1.27
Enterprise-Specific PING MIB	
jnxPingCtlIfName	1.3.6.1.4.1.2636.3.7.1.2.1.3
jnxPingCtlRoutingIfIndex	1.3.6.1.4.1.2636.3.7.1.2.1.4
jnxPingCtlRoutingIfName	1.3.6.1.4.1.2636.3.7.1.2.1.5
jnxPingCtlRoutingInstanceName	1.3.6.1.4.1.2636.3.7.1.2.1.6

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
jnxPingCtlRttThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.7
jnxPingCtlRttStdDevThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.8
jnxPingCtlRttJitterThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.9
jnxPingCtlEgressTimeThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.10
jnxPingCtlEgressStdDevThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.11
jnxPingCtlEgressJitterThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.12
jnxPingCtlIngressTimeThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.13
jnxPingCtlIngressStdDevThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.14
jnxPingCtlIngressJitterThreshold	1.3.6.1.4.1.2636.3.7.1.2.1.15
jnxPingTrapGeneration	1.3.6.1.4.1.2636.3.7.1.2.1.16
Enterprise-Specific Traceroute MIB	
jnxTRCtlIfName	1.3.6.1.4.1.2636.3.8.1.2.1.3
jnxTRCtlRoutingInstanceName	1.3.6.1.4.1.2636.3.8.1.2.1.4
RFC 3413 Target MIB	
snmpTargetSpinLock	1.3.6.1.6.3.12.1.1
snmpTargetAddrTDomain	1.3.6.1.6.3.12.1.2.1.2
snmpTargetAddrTAddress	1.3.6.1.6.3.12.1.2.1.3
snmpTargetAddrTimeout	1.3.6.1.6.3.12.1.2.1.4
snmpTargetAddrRetryCount	1.3.6.1.6.3.12.1.2.1.5
snmpTargetAddrTagList	1.3.6.1.6.3.12.1.2.1.6
snmpTargetAddrParams	1.3.6.1.6.3.12.1.2.1.7
snmpTargetAddrStorageType	1.3.6.1.6.3.12.1.2.1.8
snmpTargetAddrRowStatus	1.3.6.1.6.3.12.1.2.1.9
snmpTargetParamsMPModel	1.3.6.1.6.3.12.1.3.1.2

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
snmpTargetParamsSecurityModel	1.3.6.1.6.3.12.1.3.1.3
snmpTargetParamsSecurityLevel	1.3.6.1.6.3.12.1.3.1.4
snmpTargetParamsSecurityName	1.3.6.1.6.3.12.1.3.1.5
snmpTargetParamsStorageType	1.3.6.1.6.3.12.1.3.1.6
snmpTargetParamsRowStatus	1.3.6.1.6.3.12.1.3.1.7
RFC 3413 Notify MIB	
snmpNotifyTag	1.3.6.1.6.3.13.1.1.1.2
snmpNotifyType	1.3.6.1.6.3.13.1.1.1.3
snmpNotifyStorageType	1.3.6.1.6.3.13.1.1.1.4
snmpNotifyRowStatus	1.3.6.1.6.3.13.1.1.1.5
snmpNotifyFilterProfileName	1.3.6.1.6.3.13.1.2.1.1
snmpNotifyFilterProfileStorType	1.3.6.1.6.3.13.1.2.1.2
snmpNotifyFilterProfileRowStatus	1.3.6.1.6.3.13.1.2.1.3
snmpNotifyFilterMask	1.3.6.1.6.3.13.1.3.1.2
snmpNotifyFilterType	1.3.6.1.6.3.13.1.3.1.3
snmpNotifyFilterStorageType	1.3.6.1.6.3.13.1.3.1.4
snmpNotifyFilterRowStatus	1.3.6.1.6.3.13.1.3.1.5
RFC 2574	
usmUserSpinLock	1.3.6.1.6.3.15.1.2.1
usmUserCloneFrom	1.3.6.1.6.3.15.1.2.2.1.4
usmUserAuthProtocol	1.3.6.1.6.3.15.1.2.2.1.5
usmUserAuthKeyChange	1.3.6.1.6.3.15.1.2.2.1.6
usmUserOwnAuthKeyChange	1.3.6.1.6.3.15.1.2.2.1.7
usmUserPrivProtocol	1.3.6.1.6.3.15.1.2.2.1.8

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
usmUserPrivKeyChange	1.3.6.1.6.3.15.1.2.2.1.9
usmUserOwnPrivKeyChange	1.3.6.1.6.3.15.1.2.2.1.10
usmUserPublic	1.3.6.1.6.3.15.1.2.2.1.11
usmUserStorageType	1.3.6.1.6.3.15.1.2.2.1.12
usmUserStatus	1.3.6.1.6.3.15.1.2.2.1.13
RFC 2575	
vacmGroupName	1.3.6.1.6.3.16.1.2.1.3
vacmSecurityToGroupStorageType	1.3.6.1.6.3.16.1.2.1.4
vacmSecurityToGroupStatus	1.3.6.1.6.3.16.1.2.1.5
vacmAccessContextMatch	1.3.6.1.6.3.16.1.4.1.4
vacmAccessReadViewName	1.3.6.1.6.3.16.1.4.1.5
vacmAccessWriteViewName	1.3.6.1.6.3.16.1.4.1.6
vacmAccessNotifyViewName	1.3.6.1.6.3.16.1.4.1.7
vacmAccessStorageType	1.3.6.1.6.3.16.1.4.1.8
vacmAccessStatus	1.3.6.1.6.3.16.1.4.1.9
vacmViewSpinLock	1.3.6.1.6.3.16.1.5.1
vacmViewTreeFamilyMask	1.3.6.1.6.3.16.1.5.2.1.3
vacmViewTreeFamilyType	1.3.6.1.6.3.16.1.5.2.1.4
vacmViewTreeFamilyStorageType	1.3.6.1.6.3.16.1.5.2.1.5
vacmViewTreeFamilyStatus	1.3.6.1.6.3.16.1.5.2.1.6
RFC 2576	
snmpCommunityName	1.3.6.1.6.3.18.1.1.1.2
snmpCommunitySecurityName	1.3.6.1.6.3.18.1.1.1.3
snmpCommunityContextEngineID	1.3.6.1.6.3.18.1.1.1.4

Table 11: SNMP MIB Objects *(continued)*

Object Name	Object Identifier
snmpCommunityContextName	1.3.6.1.6.3.18.1.1.1.5
snmpCommunityTransportTag	1.3.6.1.6.3.18.1.1.1.6
snmpCommunityStorageType	1.3.6.1.6.3.18.1.1.1.7
snmpCommunityStatus	1.3.6.1.6.3.18.1.1.1.8
RFC 2576	
snmpTargetAddrMask	1.3.6.1.6.3.18.1.2.1.1
snmpTargetAddrMMS	1.3.6.1.6.3.18.1.2.1.2

Related Documentation

- Standard SNMP MIBs Supported by Junos OS on page 13
- Juniper Networks Enterprise-Specific MIBs on page 32
- Enterprise-Specific MIBs and Supported Devices on page 53

Juniper Networks Enterprise-Specific SNMP Traps

Supported Platforms LN Series, SRX Series

This topic provides pointers to the enterprise-specific SNMP traps supported by the Junos OS.

NOTE: All enterprise-specific SNMP traps supported by the Junos OS can be sent in version 1, 2, and 3 formats.

- Juniper Networks Enterprise-Specific SNMP Version 1 Traps on page 81
- Juniper Networks Enterprise-Specific SNMP Version 2 Traps on page 88
- *Juniper Networks Enterprise-Specific BGP Traps*
- *Juniper Networks Enterprise-Specific DOM Traps*
- *Juniper Networks Enterprise-Specific LDP Traps*
- *Juniper Networks Enterprise-Specific License MIB Notifications*
- *Juniper Networks Enterprise-Specific MIMSTP Traps*
- *Juniper Networks Enterprise-Specific MPLS Traps*

NOTE: For scalability reasons, the MPLS traps are generated by the ingress router only. For information about disabling the generation of MPLS traps, see the Junos OS MPLS Applications Library for Routing Devices.

- *Juniper Networks Enterprise-Specific Traps on EX Series Switches*
- *Juniper Networks Enterprise-Specific Traps on MX Series 3D Universal Edge Routers*

Related Documentation

- Standard SNMP Traps Supported on Devices Running Junos OS on page 95
- Juniper Networks Enterprise-Specific MIBs on page 32
- Standard SNMP MIBs Supported by Junos OS on page 13
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Managing Traps and Informs on page 213

Juniper Networks Enterprise-Specific SNMP Version 1 Traps

Supported Platforms vSRX

The Junos OS supports enterprise-specific SNMP version 1 traps shown in Table 12 on page 81. The traps are organized first by trap category and then by trap name. The system logging severity levels are listed for those traps that have them. Traps that do not have corresponding system logging severity levels are marked with an en dash (–).

For more information about system log messages, see the *Junos OS System Log Messages Reference*. To view the Juniper Networks enterprise-specific SNMP version 2 traps, see "Juniper Networks Enterprise-Specific SNMP Version 2 Traps" on page 88. For more information about chassis traps, see *Chassis Traps*.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
Chassis Notifications (Alarm Conditions)							
Chassis MIB (**jnx-chassis.mib**)	**jnxPowerSupplyFailure**	1.3.6.1.4.1.2636.4.1	6	1	Warning	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFanFailure	1.3.6.1.4.1.2636.4.1	6	2	Critical	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxOverTemperature	1.3.6.1.4.1.2636.4.1	6	3	Alert	CHASSISD_SNMP_TRAP	All devices running Junos OS.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
	jnxRedundancySwitchOver	1.3.6.1.4.1.2636.4.1	6	4	Critical	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruRemoval	1.3.6.1.4.1.2636.4.1	6	5	Notice	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruInsertion	1.3.6.1.4.1.2636.4.1	6	6	Notice	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruPowerOff	1.3.6.1.4.1.2636.4.1	6	7	Notice	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruPowerOn	1.3.6.1.4.1.2636.4.1	6	8	Notice	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruFailed	1.3.6.1.4.1.2636.4.1	6	9	Warning	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruOffline	1.3.6.1.4.1.2636.4.1	6	10	Notice	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruOnline	1.3.6.1.4.1.2636.4.1	6	11	Notice	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFruCheck	1.3.6.1.4.1.2636.4.1	6	12	Warning	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFEBSwitchover	1.3.6.1.4.1.2636.4.1	6	13	Warning	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxHardDiskFailed	1.3.6.1.4.1.2636.4.1	6	14	Warning	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxHardDiskMissing	1.3.6.1.4.1.2636.4.1	6	15	Warning	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
	jnxPowerSupplyOk	1.3.6.1.4.1.2636.4.2	6	1	Critical	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxFanOK	1.3.6.1.4.1.2636.4.2	6	2	Critical	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
	jnxTemperatureOK	1.3.6.1.4.1.2636.4.2	6	3	Alert	CHASSISD_ SNMP_ TRAP	All devices running Junos OS.
Configuration Notifications							
Configuration Management MIB (**jnx-configmgmt.mib**)	**jnxCmCfgChange**	1.3.6.1.4.1.2636.4.5	6	1	–	–	All devices running Junos OS.
	jnxCmRescueChange	1.3.6.1.4.1.2636.4.5	6	2	–	–	All devices running Junos OS.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
Link Notifications							
Flow Collection Services MIB (**jnx-coll.mib**)	**jnxCollUnavailableDest**	1.3.6.1.4.1.2636.4.8	6	1	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollUnavailableDestCleared	1.3.6.1.4.1.2636.4.8	6	2	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollUnsuccessfulTransfer	1.3.6.1.4.1.2636.4.8	6	3	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollFlowOverload	1.3.6.1.4.1.2636.4.8	6	4	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollFlowOverloadCleared	1.3.6.1.4.1.2636.4.8	6	5	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollMemoryUnavailable	1.3.6.1.4.1.2636.4.8	6	6	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollMemoryAvailable	1.3.6.1.4.1.2636.4.8	6	7	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollFtpSwitchover	1.3.6.1.4.1.2636.4.8	6	8	–	–	Devices that run Junos OS and have collector PICs installed.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
Passive Monitoring MIB (jnx-pmon.mib)	jnxPMonOverloadSet	1.3.6.1.4.1.2636.4.7.0.1	6	1	–	–	Devices that run Junos OS and have PICs that support passive monitoring installed.
	jnxPMonOverloadCleared	1.3.6.1.4.1.2636.4.7.0.2	6	2	–	–	Devices that run Junos OS and have PICs that support passive monitoring installed.
SONET APS MIB (jnx-sonetaps.mib)	apsEventChannelMismatch	1.3.6.1.4.1.2636.3.24.2	6	3	–	–	Devices that run Junos OS and have SONET PICs installed.
	apsEventPSBF	1.3.6.1.4.1.2636.3.24.2	6	4	–	–	Devices that run Junos OS and have SONET PICs installed.
	apsEventFEPLF	1.3.6.1.4.1.2636.3.24.2	6	5	–	–	Devices that run Junos OS and have SONET PICs installed.
Remote Operations							
PING MIB (jnx-ping.mib)	jnxPingRttThresholdExceeded	1.3.6.1.4.1.2636.4.9	6	1	–	–	All devices running Junos OS.
	jnxPingRttStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	2	–	–	All devices running Junos OS.
	jnxPingRttJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	3	–	–	All devices running Junos OS.
	jnxPingEgressThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	4	–	–	All devices running Junos OS.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
	jnxPingEgressStdDev ThresholdExceeded	1.3.6.1.4.1.2636.4.9	6	5	–	–	All devices running Junos OS.
	jnxPingEgressJitter ThresholdExceeded	1.3.6.1.4.1.2636.4.9	6	6	–	–	All devices running Junos OS.
	jnxPingIngressThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	7	–	–	All devices running Junos OS.
	jnxPingIngressStddevThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	8	–	–	All devices running Junos OS.
	jnxPingIngressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	9	–	–	All devices running Junos OS.
Routing Notifications							
BFD Experimental MIB (**jnx-bfd-exp.mib**)	**bfdSessUp**	1.3.6.1.4.1. 2636.5.3.1	6	1	–	–	All devices running Junos OS.
	bfdSessDown	1.3.6.1.4.1. 2636.5.3.1	6	2	–	–	All devices running Junos OS.
LDP MIB (**jnx-ldp.mib**)	**jnxLdpLspUp**	1.3.6.1.4.1.2636.4.4	6	1	–	–	M, T, and MX Series routers.
	jnxLdpLspDown	1.3.6.1.4.1.2636.4.4	6	2	–	–	M, T, and MX Series routers.
	jnxLdpSesUp	1.3.6.1.4.1.2636.4.4	6	3	–	–	M, T, and MX Series routers.
	jnxLdpSesDown	1.3.6.1.4.1.2636.4.4	6	4	–	–	M, T, and MX Series routers.

Table 12: Juniper Networks Enterprise-Specific Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag	Supported On
MPLS MIB (**jnx-mpls.mib**)	**mplsLspUp** (Deprecated)	1.3.6.1.4.1.2636.3.2.4	6	1	–	–	
	mplsLspDown (Deprecated)	1.3.6.1.4.1.2636.3.2.4	6	2	–	–	
	mplsLspChange (Deprecated)	1.3.6.1.4.1.2636.3.2.4	6	3	–	–	
	mplsLspPathDown (Deprecated)	1.3.6.1.4.1.2636.3.2.4	6	4	–	–	
VPN MIB (**jnx-vpn.mib**)	**jnxVpnIfUp**	1.3.6.1.4.1.2636.3.26	6	1	–	–	M, T, and MX Series routers.
	jnxVpnIfDown	1.3.6.1.4.1.2636.3.26	6	2	–	–	M, T, and MX Series routers.
	jnxVpnPwUp	1.3.6.1.4.1.2636.3.26	6	3	–	–	M, T, and MX Series routers.
	jnxVpnPwDown	1.3.6.1.4.1.2636.3.26	6	4	–	–	M, T, and MX Series routers.
RMON Alarms							
RMON MIB (**jnx-rmon.mib**)	**jnxRmonAlarmGetFailure**	1.3.6.1.4.1.2636.4.3	6	1	–	–	All devices running Junos OS.
	jnxRmonGetOk	1.3.6.1.4.1.2636.4.3	6	2	–	–	All devices running Junos OS.
SONET Alarms							
SONET MIB (**jnx-sonet.mib**)	**jnxSonetAlarmSet**	1.3.6.1.4.1.2636.4.6	6	1	–	–	Devices that run Junos OS and have SONET PICs installed.
	jnxSonetAlarmCleared	1.3.6.1.4.1.2636.4.6	6	2	–	–	Devices that run Junos OS and have SONET PICs installed.

Related Documentation

- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Standard SNMP Traps Supported on Devices Running Junos OS on page 95
- Juniper Networks Enterprise-Specific MIBs on page 32

- Standard SNMP MIBs Supported by Junos OS on page 13
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Managing Traps and Informs on page 213

Juniper Networks Enterprise-Specific SNMP Version 2 Traps

Supported Platforms vSRX

The Junos OS supports the enterprise-specific SNMP version 2 traps shown in Table 13 on page 88. The traps are organized first by trap category and then by trap name. The system logging severity levels are listed for those traps that have them. Traps that do not have corresponding system logging severity levels are marked with an en dash (–).

For more information about system messages, see the *Junos OS System Log Messages Reference*. For more information about configuring system logging, see the Junos OS Administration Library for Routing Devices. To view the Juniper Networks enterprise-specific SNMP version 1 traps, see "Juniper Networks Enterprise-Specific SNMP Version 1 Traps" on page 81. For more information about chassis traps, see *Chassis Traps*.

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
Chassis (Alarm Conditions) Notifications					
Chassis MIB (**jnx-chassis.mib**)	**jnxPowerSupplyFailure**	1.3.6.1.4.1.2636.4.1.1	Alert	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFanFailure	1.3.6.1.4.1.2636.4.1.2	Critical	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxOverTemperature	1.3.6.1.4.1.2636.4.1.3	Critical	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruNotifAdminStatus		Notice		
	jnxFruNotifMismatch		Notice		
	jnxFruNotifOperStatus		Notice		
	jnxRedundancySwitchOver	1.3.6.1.4.1.2636.4.1.4	Critical	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruRemoval	1.3.6.1.4.1.2636.4.1.5	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
	jnxFruInsertion	1.3.6.1.4.1.2636.4.1.6	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruPowerOff	1.3.6.1.4.1.2636.4.1.7	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruPowerOn	1.3.6.1.4.1.2636.4.1.8	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruFailed	1.3.6.1.4.1.2636.4.1.9	Warning	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruOffline	1.3.6.1.4.1.2636.4.1.10	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruOnline	1.3.6.1.4.1.2636.4.1.11	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFruCheck	1.3.6.1.4.1.2636.4.1.12	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxFEBSwitchover	1.3.6.1.4.1.2636.4.1.13	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxHardDiskFailed	1.3.6.1.4.1.2636.4.1.14	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
	jnxHardDiskMissing	1.3.6.1.4.1.2636.4.1.15	Notice	CHASSISD_SNMP_TRAP	All devices running Junos OS.
jnxPowerSupplyOK	1.3.6.1.4.1.2636.4.2.1	Critical	CHASSISD_SNMP_TRAP	All devices running Junos OS.	
jnxFanOK	1.3.6.1.4.1.2636.4.2.2	Critical	CHASSISD_SNMP_TRAP	All devices running Junos OS.	
jnxTemperatureOK	1.3.6.1.4.1.2636.4.2.3	Alert	CHASSISD_SNMP_TRAP	All devices running Junos OS.	
Configuration Notifications					

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
Configuration Management MIB (**jnx-cfgmgmt.mib**)	**jnxCmCfgChange**	1.3.6.1.4.1.2636.4.5.0.1	–	–	All devices running Junos OS.
	jnxCmRescueChange	1.3.6.1.4.1.2636.4.5.0.2	–	–	All devices running Junos OS.
Link Notifications					
Flow Collection Services MIB (**jnx-coll.mib**)	**jnxCollUnavailableDest**	1.3.6.1.4.1.2636.4.8.0.1	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollUnavailableDestCleared	1.3.6.1.4.1.2636.4.8.0.2	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollUnsuccessfulTransfer	1.3.6.1.4.1.2636.4.8.0.3	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollFlowOverload	1.3.6.1.4.1.2636.4.8.0.4	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollFlowOverloadCleared	1.3.6.1.4.1.2636.4.8.0.5	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollMemoryUnavailable	1.3.6.1.4.1.2636.4.8.0.6	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollMemoryAvailable	1.3.6.1.4.1.2636.4.8.0.7	–	–	Devices that run Junos OS and have collector PICs installed.
	jnxCollFtpSwitchover	1.3.6.1.4.1.2636.4.8.0.8	–	–	Devices that run Junos OS and have collector PICs installed.
PMON MIB (**jnx-pmon.mib**)	**jnxPMonOverloadSet**	1.3.6.1.4.1.2636.4.7.0.1	–	–	Devices that run Junos OS and have PICs that support passive monitoring installed.
	jnxPMonOverloadCleared	1.3.6.1.4.1.2636.4.7.0.2	–	–	Devices that run Junos OS and have PICs that support passive monitoring installed.

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
SONET APS MIB (**jnx-sonetaps.mib**)	**apsEventChannelMismatch**	1.3.6.1.4.1.2636.3.24.2.0.3	–	–	Devices that run Junos OS and have SONET PICs installed.
	apsEventPSBF	1.3.6.1.4.1.2636.3.24.2.0.4	–	–	Devices that run Junos OS and have SONET PICs installed.
	apsEventFEPLF	1.3.6.1.4.1.2636.3.24.2.0.5	–	–	Devices that run Junos OS and have SONET PICs installed.
Remote Operations Notifications					
PING MIB (**jnx-ping.mib**)	**jnxPingRttThreshold Exceeded**	1.3.6.1.4.1.2636.4.9.0.1	–	–	All devices running Junos OS.
	jnxPingRttStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.2	–	–	All devices running Junos OS.
	jnxPingRttJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.3	–	–	All devices running Junos OS.
	jnxPingEgressThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.4	–	–	All devices running Junos OS.
	jnxPingEgressStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.5	–	–	All devices running Junos OS.
	jnxPingEgressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.6	–	–	All devices running Junos OS.
	jnxPingIngressThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.7	–	–	All devices running Junos OS.
	jnxPingIngressStddevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.8	–	–	All devices running Junos OS.
	jnxPingIngressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.9	–	–	All devices running Junos OS.
Routing Notifications					
BFD Experimental MIB (**jnx-bfd-exp.mib**)	**bfdSessUp**	1.3.6.1.4.1.2636.5.3.1.0.1	–	–	All devices running Junos OS.
	bfdSessDown	1.3.6.1.4.1.2636.5.3.1.0.2	–	–	All devices running Junos OS.

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
BGP4 V2 MIB (**jnx-bgpmib2.mib**)	**jnxBgpM2Established**	1.3.6.1.4.1.2636.5.1.1.1.0.1	–	–	All devices running Junos OS.
	jnxBgpM2BackwardTransition	1.3.6.1.4.1.2636.5.1.1.1.0.2	–	–	All devices running Junos OS.
DHCP MIB (**jnx-dhcp.mib**)	**jnxJdhcpLocalServer DuplicateClient**	1.3.6.1.4.1.2636.3.61.61.1.3.1	–	–	All devices running Junos OS.
	jnxJdhcpLocalServer InterfaceLimitExceeded	1.3.6.1.4.1.2636.3.61.61.1.3.2	–	–	All devices running Junos OS.
	jnxJdhcpLocalServer InterfaceLimitAbated	1.3.6.1.4.1.2636.3.61.61.1.3.3	–	–	All devices running Junos OS.
	jnxJdhcpLocalServer Health	1.3.6.1.4.1.2636.3.61.61.1.3.4	–	–	All devices running Junos OS.
	jnxJdhcpRelayInterface LimitExceeded	1.3.6.1.4.1.2636.3.61.61.2.3.1	–	–	All devices running Junos OS.
	jnxJdhcpRelayInterface LimitAbated	1.3.6.1.4.1.2636.3.61.61.2.3.2	–	–	All devices running Junos OS.
DHCPv6MIB (**jnx-dhcpv6.mib**)	**jnxJdhcpv6LocalServer InterfaceLimitExceeded**	1.3.6.1.4.1.2636.3.62.62.2.3.1	–	–	All devices running Junos OS.
	jnxJdhcpv6LocalServer InterfaceLimitAbated	1.3.6.1.4.1.2636.3.62.62.2.3.2	–	–	All devices running Junos OS.
	jnxJdhcpv6LocalServer Health	1.3.6.1.4.1.2636.3.62.62.2.3.3	–	–	All devices running Junos OS.
LDP MIB (**jnx-ldp.mib**)	**jnxLdpLspUp**	1.3.6.1.4.1.2636.4.4.0.1	–	–	M, T, and MX Series routers.
	jnxLdpLspDown	1.3.6.1.4.1.2636.4.4.0.2	–	–	M, T, and MX Series routers.
	jnxLdpSesUp	1.3.6.1.4.1.2636.4.4.0.3	–	–	M, T, and MX Series routers.
	jnxLdpSesDown	1.3.6.1.4.1.2636.4.4.0.4	–	–	M, T, and MX Series routers.

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
MPLS MIB (**jnx-mpls.mib**)	**mplsLspUp** (Deprecated)	1.3.6.1.4.1.2636.3.2.4.1	–	–	
	mplsLspInfoUp	1.3.6.1.4.1.2636.3.2.0.1	–	–	M, T, and MX Series routers.
	mplsLspDown (Deprecated)	1.3.6.1.4.1.2636.3.2.4.2	–	–	
	mplsLspInfoDown	1.3.6.1.4.1.2636.3.2.0.2	–	–	M, T, and MX Series routers.
	mplsLspChange (Deprecated)	1.3.6.1.4.1.2636.3.2.4.3	–	–	
	mplsLspInfoChange	1.3.6.1.4.1.2636.3.2.0.3	–	–	M, T, and MX Series routers.
	mplsLspPathDown (Deprecated)	1.3.6.1.4.1.2636.3.2.4.4	–	–	
	mplsLspInfoPathDown	1.3.6.1.4.1.2636.3.2.0.4	–	–	M, T, and MX Series routers.
	mplsLspInfoPathUp	1.3.6.1.4.1.2636.3.2.0.5	–	–	M, T, and MX Series routers.
VPN MIB (**jnx-vpn.mib**)	**jnxVpnIfUp**	1.3.6.1.4.1.2636.3.26.0.1	–	–	M, T, and MX Series routers.
	jnxVpnIfDown	1.3.6.1.4.1.2636.3.26.0.2	–	–	M, T, and MX Series routers.
	jnxVpnPwUp	1.3.6.1.4.1.2636.3.26.0.3	–	–	M, T, and MX Series routers.
	jnxVpnPwDown	1.3.6.1.4.1.2636.3.26.0.4	–	–	M, T, and MX Series routers.

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
AAA MIB (**jnx-user-aaa.mib**)	**jnxAccessAuthAddress PoolHighThreshold**	1.3.6.1.4.1.2636.3.51.1.0.5	–	–	SRX Series devices.
	jnxAccessAuthAddress PoolAbateThreshold	1.3.6.1.4.1.2636.3.51.1.0.6	–	–	SRX Series devices.
	jnxAccessAuthAddress PoolOutOfAddresses	1.3.6.1.4.1.2636.3.51.1.0.7	–	–	SRX Series devices.
	jnxAccessAuthAddress PoolOutOfMemory	1.3.6.1.4.1.2636.3.51.1.0.8	–	–	SRX Series devices.
	jnxAccessAuthService Up	1.3.6.1.4.1.2636.3.51. 1.0.1	–	–	SRX Series devices.
	jnxAccessAuthService Down	1.3.6.1.4.1.2636.3.51. 1.0.2	–	–	SRX Series devices.
	jnxAccessAuthServer Disabled	1.3.6.1.4.1.2636.3.51. 1.0.3	–	–	SRX Series devices.
	jnxAccessAuthServer Enabled	1.3.6.1.4.1.2636.3.51. 1.0.4	–	–	SRX Series devices.
	jnxJsFwAuthFailure	1.3.6.1.4.1.2636.3.39.1.2. 1.0.1	–	–	SRX Series devices.
Access Authentication Methods MIB (**jnx-js-auth. mib**)	**jnxJsFwAuthServiceUp**	1.3.6.1.4.1.2636.3.39.1.2. 1.0.2	–	–	SRX Series devices.
	jnxJsFwAuthServiceDown	1.3.6.1.4.1.2636.3.39.1.2. 1.0.3	–	–	SRX Series devices.
	jnxJsFwAuthCapacityExceeded	1.3.6.1.4.1.2636.3.39.1.2. 1.0.4	–	–	SRX Series devices.
	jnxJsNatAddrPool ThresholdStatus	1.3.6.1.4.1.2636.3.39.1.7. 1.0.1	–	–	SRX Series devices.
Network Address Translation Resources-Monitoring MIB (**jnxNatMIB**)	**jnxNatAddrPoolUtil**	1.3.6.1.4.1.2636.3.59.1.2.1	–	–	M Series and MX Series routers
	jnxNatTrapSrcPoolName	1.3.6.1.4.1.2636.3.59.1.2.2	–	–	M Series and MX Series routers
	jnxNatAddrPoolThresholdStatus	1.3.6.1.4.1.2636.3.59.1.0.1	–	–	M Series and MX Series routers

Table 13: Juniper Networks Enterprise-Specific Supported SNMP Version 2 Traps *(continued)*

Source MIB	Trap Name	snmpTrapOID	System Logging Severity Level	System Log Tag	Supported On
Network Address Translation MIB (**jnx-js-nat.mib**)	**jnxJsScreen Attack**	1.3.6.1.4.1.2636.3.39.1.8.1.0.1	Warning	RT_SCREEN_ICMP, RT_SCREEN_IP, RT_SCREEN_SESSION_LIMIT, RT_SCREEN_TCP, RT_SCREEN_UDP	SRX Series devices.
Security Screening Objects MIB (**jnx-js-screening.mib**)	**jnxJsScreenCfg Change**	1.3.6.1.4.1.2636.3.39.1.8.1.0.2	–	–	SRX Series devices.
RMON Alarms					
RMON MIB (**jnx-rmon.mib**)	**jnxRmonGetOk**	1.3.6.1.4.1.2636.4.3.0.2	–	–	All devices running Junos OS.
SONET Alarms					
SONET MIB (**jnx-sonet.mib**)	**jnxSonetAlarm Cleared**	1.3.6.1.4.1.2636.4.6.0.2	–	–	Devices that run Junos OS and have SONET PICs installed.

Related Documentation

- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Standard SNMP Traps Supported on Devices Running Junos OS on page 95
- Juniper Networks Enterprise-Specific MIBs on page 32
- Standard SNMP MIBs Supported by Junos OS on page 13
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Managing Traps and Informs on page 213

Standard SNMP Traps Supported on Devices Running Junos OS

Supported Platforms LN Series, SRX Series

This topic provides pointers to the standard SNMP traps supported by the Junos OS.

NOTE: For scalability reasons, the MPLS traps are generated by the ingress router only.

- Standard SNMP Version 1 Traps on page 96
- Standard SNMP Version 2 Traps on page 99
- *Standard SNMP Traps on EX Series Ethernet Switches*
- Unsupported Standard SNMP Traps on page 106

Related Documentation

- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Juniper Networks Enterprise-Specific MIBs on page 32
- Standard SNMP MIBs Supported by Junos OS on page 13
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Managing Traps and Informs on page 213

Standard SNMP Version 1 Traps

Supported Platforms LN Series, SRX Series

Table 14 on page 96 provides an overview of the standard traps for SNMPv1. The traps are organized first by trap category and then by trap name, and include their enterprise ID, generic trap number, and specific trap number. The system logging severity levels are listed for those traps that have them with their corresponding system log tag. Traps that do not have corresponding system logging severity levels are marked with an en dash (–) in the table.

For more information about system log messages, see the *Junos OS System Log Messages Reference*. For more information about configuring system logging, see the *Junos OS System Basics Configuration Guide*.

Table 14: Standard Supported SNMP Version 1 Traps

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag	Supported On
Startup Notifications							
RFC 1215, *Conventions for Defining Traps for Use with the SNMP*	**authenticationFailure**	1.3.6.1.4.1.2636	4	0	Notice	SNMPD_ TRAP_ GEN_FAILURE	All devices running Junos OS.
	coldStart	1.3.6.1.4.1.2636	0	0	Critical	SNMPD_TRAP_ COLD_START	All devices running Junos OS.
	warmStart	1.3.6.1.4.1.2636	1	0	Error	SNMPD_TRAP_ WARM_START	All devices running Junos OS.
Link Notifications							

Table 14: Standard Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag	Supported On
RFC 1215, *Conventions for Defining Traps for Use with the SNMP*	**linkDown**	1.3.6.1.4.1.2636	2	0	Warning	SNMP_ TRAP_ LINK_DOWN	All devices running Junos OS.
	linkUp	1.3.6.1.4.1.2636	3	0	Info	SNMP_TRAP_ LINK_UP	All devices running Junos OS.
Remote Operations Notifications							
RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*	**pingProbeFailed**	1.3.6.1.2.1.80.0	6	1	Info	SNMP_TRAP _PING_ PROBE_ FAILED	All devices running Junos OS.
	pingTestFailed	1.3.6.1.2.1.80.0	6	2	Info	SNMP_TRAP_ PING_TEST _FAILED	All devices running Junos OS.
	pingTestCompleted	1.3.6.1.2.1.80.0	6	3	Info	SNMP_TRAP_ PING_TEST_ COMPLETED	All devices running Junos OS.
	traceRoutePathChange	1.3.6.1.2.1.81.0	6	1	Info	SNMP_TRAP_ TRACE_ROUTE_ PATH_CHANGE	All devices running Junos OS.
	traceRouteTestFailed	1.3.6.1.2.1.81.0	6	2	Info	SNMP_TRAP_ TRACE_ROUTE_ TEST_FAILED	All devices running Junos OS.
	traceRouteTestCompleted	1.3.6.1.2.1.81.0	6	3	Info	SNMP_TRAP_ TRACE_ROUTE_ TEST_COMPLETED	All devices running Junos OS.
RMON Alarms							
RFC 2819a, *RMON MIB*	**fallingAlarm**	1.3.6.1.2.1.16	6	2	–	–	All devices running Junos OS.
	risingAlarm	1.3.6.1.2.1.16	6	1	–	–	All devices running Junos OS.
Routing Notifications							
BGP 4 MIB	**bgpEstablished**	1.3.6.1.2.1.15.7	6	1	–	–	M, T, MX, J, EX, and SRX for branch devices.
	bgpBackwardTransition	1.3.6.1.2.1.15.7	6	2	–	–	M, T, MX, J, EX, and SRX for branch devices.

Table 14: Standard Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag	Supported On
OSPF TRAP MIB	**ospfVirtIfStateChange**	1.3.6.1.2.1.14.16.2	6	1	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfNbrStateChange	1.3.6.1.2.1.14.16.2	6	2	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2	6	3	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfIfConfigError	1.3.6.1.2.1.14.16.2	6	4	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2	6	5	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfIfAuthFailure	1.3.6.1.2.1.14.16.2	6	6	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfVirtIfAuthFailure	1.3.6.1.2.1.14.16.2	6	7	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	8	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	9	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfTxRetransmit	1.3.6.1.2.1.14.16.2	6	10	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2	6	11	–	–	M, T, MX, J, EX, and SRX for branch devices.
	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2	6	13	–	–	M, T, MX, J, EX, and SRX for branch devices.

Table 14: Standard Supported SNMP Version 1 Traps *(continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag	Supported On
	ospfIfStateChange	1.3.6.1.2.1.14.16.2	6	16	–	–	M, T, MX, J, EX, and SRX for branch devices.
VRRP Notifications							
RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*	**vrrpTrapNewMaster**	1.3.6.1.2.1.68	6	1	Warning	VRRPD_NEW MASTER_TRAP	All devices running Junos OS.
	vrrpTrapAuthFailure	1.3.6.1.2.1.68	6	2	Warning	VRRPD_AUTH_ FAILURE_TRAP	All devices running Junos OS.
RFC 6527, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol Version 3 (VRRPv3)*	**vrrpv3NewMaster**	1.3.6.1.2.1.207	6	1	Warning	VRRPD_NEW_MASTER	M and MX
	vrrpv3ProtoError	1.3.6.1.2.1.207	6	2	Warning	VRRPD_V3_PROTO_ERROR	M and MX

Related Documentation

- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Standard SNMP Traps Supported on Devices Running Junos OS on page 95
- Juniper Networks Enterprise-Specific MIBs on page 32
- Standard SNMP MIBs Supported by Junos OS on page 13
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Managing Traps and Informs on page 213

Standard SNMP Version 2 Traps

Supported Platforms ACX Series, M Series, MX Series, PTX Series, SRX Series, T Series

Table 15 on page 100 provides an overview of the standard SNMPv2 traps supported by the Junos OS. The traps are organized first by trap category and then by trap name and include their **snmpTrapOID**. The system logging severity levels are listed for those traps that have them with their corresponding system log tag. Traps that do not have corresponding system logging severity levels are marked with an en dash (–) in the table.

For more information about system log messages, see *System Log Monitoring and Troubleshooting Guide for Security Devices*.

Table 15: Standard Supported SNMP Version 2 Traps

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
Startup Notifications					
RFC 1907, *Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)*	**coldStart**	1.3.6.1.6.3.1.1.5.1	Critical	SNMPD_TRAP_COLD_START	All devices running Junos OS.
	warmStart	1.3.6.1.6.3.1.1.5.2	Error	SNMPD_TRAP_WARM_START	All devices running Junos OS.
	authenticationFailure	1.3.6.1.6.3.1.1.5.5	Notice	SNMPD_TRAP_GEN_FAILURE	All devices running Junos OS.
Link Notifications					
RFC 2863, *The Interfaces Group MIB*	**linkDown**	1.3.6.1.6.3.1.1.5.3	Warning	SNMP_TRAP_LINK_DOWN	All devices running Junos OS.
	linkUp	1.3.6.1.6.3.1.1.5.4	Info	SNMP_TRAP_LINK_UP	All devices running Junos OS.
Remote Operations Notifications					
RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*	**pingProbeFailed**	1.3.6.1.2.1.80.0.1	Info	SNMP_TRAP_PING_PROBE_FAILED	All devices running Junos OS.
	pingTestFailed	1.3.6.1.2.1.80.0.2	Info	SNMP_TRAP_PING_TEST_FAILED	All devices running Junos OS.
	pingTestCompleted	1.3.6.1.2.1.80.0.3	Info	SNMP_TRAP_PING_TEST_COMPLETED	All devices running Junos OS.
	traceRoutePathChange	1.3.6.1.2.1.81.0.1	Info	SNMP_TRAP_TRACE_ROUTE_PATH_CHANGE	All devices running Junos OS.
	traceRouteTestFailed	1.3.6.1.2.1.81.0.2	Info	SNMP_TRAP_TRACE_ROUTE_TEST_FAILED	All devices running Junos OS.
	traceRouteTestCompleted	1.3.6.1.2.1.81.0.3	Info	SNMP_TRAP_TRACE_ROUTE_TEST_COMPLETED	All devices running Junos OS.
RMON Alarms					

Table 15: Standard Supported SNMP Version 2 Traps *(continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
RFC 2819a, *RMON MIB*	**fallingAlarm**	1.3.6.1.2.1.16.0.1	–	–	All devices running Junos OS.
	risingAlarm	1.3.6.1.2.1.16.0.2	–	–	All devices running Junos OS.
Routing Notifications					
BGP 4 MIB	**bgpEstablished**	1.3.6.1.2.1.15.7.1	–	–	All devices running Junos OS.
	bgpBackwardTransition	1.3.6.1.2.1.15.7.2	–	–	All devices running Junos OS.

Table 15: Standard Supported SNMP Version 2 Traps *(continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
OSPF Trap MIB	**ospfVirtIfStateChange**	1.3.6.1.2.1.14.16.2.1	–	–	All devices running Junos OS.
	ospfNbrStateChange	1.3.6.1.2.1.14.16.2.2	–	–	All devices running Junos OS.
	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2.3	–	–	All devices running Junos OS.
	ospfIfConfigError	1.3.6.1.2.1.14.16.2.4	–	–	All devices running Junos OS.
	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2.5	–	–	All devices running Junos OS.
	ospfIfAuthFailure	1.3.6.1.2.1.14.16.2.6	–	–	All devices running Junos OS.
	ospfVirtIfAuthFailure	1.3.6.1.2.1.14.16.2.7	–	–	All devices running Junos OS.
	ospfIfRxBadPacket	1.3.6.1.2.1.14.16.2.8	–	–	All devices running Junos OS.
	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2.9	–	–	All devices running Junos OS.
	ospfTxRetransmit	1.3.6.1.2.1.14.16.2.10	–	–	All devices running Junos OS.
	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2.11	–	–	All devices running Junos OS.
	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2.13	–	–	All devices running Junos OS.
	ospfIfStateChange	1.3.6.1.2.1.14.16.2.16	–	–	All devices running Junos OS.
MPLS Notifications					

Table 15: Standard Supported SNMP Version 2 Traps *(continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
RFC 3812, *Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base*	**mplsTunnelUp**				
	mplsTunnelDown				
	mplsTunnelRerouted				
	mplsTunnelReoptimized				
Entity State MIB Notifications					
RFC 4268, *Entity State MIB*	**entStateOperEnabled**	1.3.6.1.2.1.131.0.1	Notice	CHASSISD_SNMP_TRAP3	MX240, MX480, and MX960
	entStateOperDisabled	1.3.6.1.2.1.131.0.2	Notice	CHASSISD_SNMP_TRAP3	MX240, MX480, and MX960
L3VPN Notifications					
RFC 4382, *MPLS/BGP Layer 3 Virtual Private Network (VPN)*	**mplsL3VpnVrfUp**				
	mplsL3VpnVrfDown				
	mplsL3VpnVrfRouteMidThreshExceeded				
	mplsL3VpnNumVrfRouteMaxThreshExceeded				
	mplsL3VpnNumVrfRouteMaxThreshCleared				
VRRP Notifications					
RFC 2787, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol*	**vrrpTrapNewMaster**	1.3.6.1.2.1.68.0.1	Warning	VRRPD_ NEWMASTER_TRAP	All devices running Junos OS.
	vrrpTrapAuthFailure	1.3.6.1.2.1.68.0.2	Warning	VRRPD_AUTH_ FAILURE_ TRAP	All devices running Junos OS.
RFC 6527, *Definitions of Managed Objects for the Virtual Router Redundancy Protocol Version 3 (VRRPv3)*	**vrrpv3NewMaster**	1.3.6.1.2.1.207.0.1	Warning	VRRPD_NEW_MASTER	M and MX
	vrrpv3ProtoError	1.3.6.1.2.1.207.0.2	Warning	VRRPD_V3_PROTO_ERROR	M and MX

Table 15: Standard Supported SNMP Version 2 Traps *(continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On

The Junos OS also supports the following standard SNMP version 2 traps:

- SNMP Version 2 MPLS Traps on page 104
- SNMP Version 2 L3VPN Traps on page 105

SNMP Version 2 MPLS Traps

The Junos OS supports the MPLS SNMP version 2 traps defined in RFC 3812, *Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base*.

You can disable the MPLS traps by including the **no-trap** option at the **[edit protocol mpls log-updown]** hierarchy level. For information about disabling the generation of MPLS traps, see the *Junos OS MPLS Applications Configuration Guide*.

The Junos OS supports the following MPLS traps:

- **mplsTunnelUp**—Generated when an **mplsTunnelOperStatus** object for one of the configured tunnels leaves the **down** state and transitions into another state, other than the **notPresent** state.
- **mplsTunnelDown**—Generated when an **mplsTunnelOperStatus** object for one of the configured tunnels enters the **down** state from a state other than the **notPresent** state.

  NOTE: When an LSP flaps, only the ingress and egress routers of that LSP generate the **mplsTunnelUp** and **mplsTunnelDown** traps. Previously, all the routers associated with an LSP—that is, the ingress, egress, and transit routers—used to generate the traps when the LSP flaps.

- **mplsTunnelRerouted**—Generated when a tunnel is rerouted.
- **mplsTunnelReoptimized**—Generated when a tunnel is reoptimized.

  NOTE: In Junos OS Release 8.3 and earlier, **mplsTunnelReoptimized** was generated every time the optimization timer expired; that is, when the optimization timer exceeded the value set for the **optimize-timer** statement at the **[edit protocols mpls label-switched-path path-name]** hierarchy level. However, in Release 8.4 and later, this trap is generated only when the path is reoptimized, and not when the optimization timer expires.

SNMP Version 2 L3VPN Traps

The Junos OS also supports the following L3VPN SNMP version 2 traps defined in RFC 4382, *MPLS/BGP Layer 3 Virtual Private Network (VPN)*:

- **mplsL3VpnVrfUp**—Generated when:
  - No interface is associated with this VRF, and the first (and only first) interface associated with it has its **ifOperStatus**change to **up**.
  - Only one interface is associated with this VRF, and the **ifOperStatus** of this interface changes to **up**.
  - Multiple interfaces are associated with this VRF, and the **ifOperStatus** of all interfaces is **down**, and the first of those interfaces has its **ifOperStatus** change to **up**.
- **mplsL3VpnVrfDown**—Generated when:
  - One interface is associated with this VRF, and the **ifOperStatus** of this interface changes from **up** to **down**.
  - Multiple interfaces are associated with this VRF, and the **ifOperStatus** of all except one of these interfaces is equal to **up**, and the **ifOperStatus** of that interface changes from **up** to **down**.
  - The last interface with **ifOperStatus** equal to **up** is disassociated from a VRF.
- **mplsL3VpnVrfRouteMidThreshExceeded**—Generated when the number of routes contained by the specified VRF exceeds the value indicated by **mplsL3VpnVrfMidRouteThreshold**.

  You can configure the **mplsL3VpnVrfMidRouteThreshold** value as follows:

  ```
  [edit routing-instances <instance-name>]
  user@R1# set routing-options maximum-paths <limit> threshold <threshold-value>
  ```

  This configuration sets the **mplsL3VpnVrfMidRouteThreshold** value to **<threshold-value>** % of **<limit>**. This value can also be calculated as (<limit>*<threshold-value>) /100.
- **mplsL3VpnVrfNumVrfRouteMaxThreshExceeded**—Generated when the number of routes contained by the specified VRF exceeds or attempts to exceed the maximum allowed value as indicated by **mplsL3VpnVrfMaxRouteThreshold**.
- **mplsL3VpnNumVrfSecIllglLblThrshExcd** —Generated when the number of illegal label violations on a VRF as indicated by **mplsL3VpnVrfSecIllegalLblVltns**has exceeded **mplsL3VpnIllLblRcvThrsh**.
- **mplsL3VpnNumVrfRouteMaxThreshCleared**—Generated only after the number of routes contained by the specified VRF exceeds or attempts to exceed the maximum allowed value as indicated by **mplsVrfMaxRouteThreshold**, and then falls below this value.

Related Documentation

- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Standard SNMP Traps Supported on Devices Running Junos OS on page 95
- Juniper Networks Enterprise-Specific MIBs on page 32

- Standard SNMP MIBs Supported by Junos OS on page 13
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Managing Traps and Informs on page 213

Unsupported Standard SNMP Traps

Supported Platforms LN Series, SRX Series

Standard SNMP traps that are defined in MIBs supported by the Junos OS but are not generated by the Junos OS are shown in Table 16 on page 107.

Table 16: Unsupported Standard SNMP Traps

MIB	Trap Name	Description
isismib.mib	isisDatabaseOverload	Generated when the system enters or leaves the overload state.
	isisManualAddressDrops	Generated when one of the manual **areaAddresses** assigned to the system is ignored when computing routes.
	isisCorruptedLSPDetected	Generated when an LSP stored in memory becomes corrupted.
	isisAttemptToExceedMaxSequence	Generated when the sequence number on a generated LSP wraps the 32-bit sequence counter and the number is purged.
	isisIDLenMismatch	Generated when a protocol data unit (PDU) is received with a different value for the system ID length. This trap includes an index to identify the circuit where the PDU was received and the PDU header.
	isisMaxAreaAddressesMismatch	Generated when a PDU with a different value for the maximum area addresses is received.
	isisOwnLSPPurge	Generated when a PDU is received with a system ID and zero age. This notification includes the circuit index if available.
	isisSequenceNumberSkip	Generated when an LSP is received with a system ID and different contents, indicating the LSP might require a higher sequence number.
	isisAuthenticationTypeFailure	Generated when a PDU with the wrong authentication type field is received.
	isisAuthenticationFailure	Generated when a PDU with an incorrect authentication information field is received.
	isisVersionSkew	Generated when a hello PDU from an IS running a different version of the protocol is received.
	isisAreaMismatch	Generated when a hello PDU from an IS which does not share any area address is received.
	isisRejectedAdjacency	Generated when a hello PDU from an IS is received, but no adjacency is established because of a lack of resources.
	isisLSPTooLargeToPropagate	Generated when a link-state PDU that is larger than the **dataLinkBlockSize** for a circuit is attempted, but not propagated.
	isisOriginatingLSPBufferSizeMismatch	

Table 16: Unsupported Standard SNMP Traps *(continued)*

MIB	Trap Name	Description
		Generated when a Level 1 link-state PDU or Level 2 link-state PDU is received that is larger than the local value for originating **L1LSPBufferSize** or originating **L2LSPBufferSize**, respectively, or when a Level 1 link-state PDU or Level 2 link-state PDU is received containing the originating **LSPBufferSize** option and the value in the PDU option field does not match the local value for originating **L1LSPBufferSize** or originating **L2LSPBufferSize**, respectively.
	isisProtocolsSupportedMismatch	Generated when a nonpseudonode, segment 0 link-state PDU is received that has no matching protocols.
l3vpnmib.mib	**mplsVrfIfUp**	Generated when the **ifOperStatus** of an interface associated with a VRF table changes to the **up(1)** state, or when an interface with **ifOperStatus = up(1)** is associated with a VRF table.
	mplsVrfIfDown	Generated when the **ifOperStatus** of an interface associated with a VRF table changes to the **down(1)** state, or when an interface with **ifOperStatus = up(1)** state is disassociated from a VRF table.
	mplsNumVrfRouteMidThreshExceeded	Generated when the number of routes contained by the specified VRF table exceeds the value indicated by **mplsVrfMidRouteThreshold**.
	mplsNumVrfRouteMaxThreshExceeded	Generated when the number of routes contained by the specified VRF table reaches or attempts to exceed the maximum allowed value as indicated by **mplsVrfMaxRouteThreshold**.
	mplsNumVrfSecIllglLblThrshExcd	Generated when the number of illegal label violations on a VRF table as indicated by **mplsVpnVrfSecIllegalLblVltns** has exceeded **mplsVpnVrfSecIllegalLblRcvThrsh**.
msdpmib.mib	**msdpEstablished**	Generated when the Multicast Source Discovery Protocol (MSDP) finite state machine (FSM) enters the **Established** state.
	msdpBackwardTransition	Generated when the MSDP FSM moves from a higher numbered state to a lower numbered state.
ospf2trap.mib	**ospfOriginateLsa**	Generated when a new LSA is originated by the router because of a topology change.
	ospfLsdbOverflow	Generated when the number of LSAs in the router's link-state database exceeds the value of **ospfExtLsdbLimit**.
	ospfLsdbApproachingOverflow	Generated when the number of LSAs in the router's link-state database exceeds 90% of the value of **ospfExtLsdbLimit**.

Table 16: Unsupported Standard SNMP Traps *(continued)*

MIB	Trap Name	Description
rfc1747.mib	sdlcPortStatusChange	Generated when the state of an SDLC port transitions to active or inactive.
	sdlcLSStatusChange	Generated when the state of an SDLC link station transitions to contacted or disconnected.
rfc2115a.mib	frDLCIStatusChange	Generated when a virtual circuit changes state (has been created or invalidated, or has toggled between the active and inactive states).
rfc2662.mib	adslAtucRateChangeTrap	Generated when the ATUCs transmit rate has changed (RADSL mode only).
	adslAtucPerfLofsThreshTrap	Generated when the loss of framing 15-minute interval threshold is reached.
	adslAtucInitFailureTrap	Generated when ATUC initialization fails.
	adslAturPerfLprsThreshTrap	Generated when the loss of power 15-minute interval threshold is reached.
	adslAturRateChangeTrap	Generated when the ATURs transmit rate changes (RADSL mode only).
rfc3020.mib	mfrMibTrapBundleLinkMismatch	Generated when a bundle link mismatch is detected.
rfc3813.mib	mplsXCUp	Generated when **mplsXCOperStatus** for one or more contiguous entries in **mplsXCTable** enters the **up(1)** state from some other state.
	mplsXCDown	Generated when **mplsXCOperStatus** for one or more contiguous entries in **mplsXCTable** enters the **down(2)** state from some other state.

Related Documentation

- Juniper Networks Enterprise-Specific SNMP Traps on page 80
- Standard SNMP Traps Supported on Devices Running Junos OS on page 95
- Juniper Networks Enterprise-Specific MIBs on page 32
- Standard SNMP MIBs Supported by Junos OS on page 13

CHAPTER 4

Loading MIB Files to a Network Management System

Loading MIB Files to a Network Management System

Supported Platforms LN Series, SRX Series

For your network management system (NMS) to identify and understand the MIB objects used by the Junos OS, you must first load the MIB files to your NMS using a MIB compiler. A MIB compiler is a utility that parses the MIB information such as the MIB object name, IDs, and data type for the NMS.

You can download the Junos MIB package from the **Enterprise-Specific MIBs and Traps** section of the Junos OS Technical Publications index page at http://www.juniper.net/techpubs/software/junos/index.html . The Junos MIB package is available in **.zip** and **.tar** packages. You can download the appropriate format based on your requirements.

The Junos MIB package contains two folders: **StandardMibs** and **JuniperMibs**. The **StandardMibs** folder contains the standard MIBs and RFCs that are supported on devices running the Junos OS, whereas the **JuniperMibs** folder contains the Juniper Networks enterprise-specific MIBs.

To load MIB files that are required for managing and monitoring devices running the Junos OS:

1. Go to the Junos OS Technical Publications index page (http://www.juniper.net/techpubs/software/junos/index.html).
2. Click the tab that corresponds to the Junos OS Release for which you want to download the MIB files.
3. On the selected tab, click the **+** (plus) sign that corresponds to the **Enterprise-Specific MIBs and Traps** section to expand the section.
4. Click the **TAR** or **ZIP** link that corresponds to the **Enterprise MIBs** link under the **Enterprise-Specific MIBs and Traps** section to download the Junos MIB package.
5. Decompress the file (**.tar** or **.zip**) using an appropriate utility.

6. Load the standard MIB files (from the **StandardMibs** folder) in the following order:

   NOTE: Some of the MIB compilers that are commonly used have the standard MIBs preloaded on them. If the standard MIBs are already loaded on the MIB compiler that you are using, skip this step and proceed to Step 7.

   a. **mib-SNMPv2-SMI.txt**
   b. **mib-SNMPv2-TC.txt**
   c. **mib-IANAifType-MIB.txt**
   d. **mib-IANA-RTPROTO-MIB.txt**
   e. **mib-rfc1907.txt**
   f. **mib-rfc2011a.txt**
   g. **mib-rfc2012a.txt**
   h. **mib-rfc2013a.txt**
   i. **mib-rfc2863a.txt**

7. Load the remaining standard MIB files.

   NOTE: You must follow the order specified in this procedure, and ensure that all standard MIBs are loaded before you load the enterprise-specific MIBs. There might be dependencies that require a particular MIB to be present on the compiler before loading some other MIB. You can find such dependencies listed in the **IMPORT** section of the MIB file.

8. Load the Juniper Networks enterprise-specific SMI MIB, **mib-jnx-smi.txt**, and the following optional SMI MIBs based on your requirements:

   - **mib-jnx-js-smi.txt**—(Optional) For Juniper Security MIB tree objects
   - **mib-jnx-ex-smi.txt**—(Optional) For EX Series Ethernet Switches
   - **mib-jnx-exp.txt**—(Recommended) For Juniper Networks experimental MIB objects

9. Load the remaining enterprise-specific MIBs from the **JuniperMibs** folder.

   TIP: While loading a MIB file, if the compiler returns an error message saying that any of the objects is undefined, open the MIB file using a text editor and ensure that all the MIB files listed in the **IMPORT** section are loaded on the compiler. If any of the MIB files listed in the **IMPORT** section is not loaded on the compiler, load that MIB file, and then try to load the MIB file that failed to load.

For example, the enterprise-specific PING MIB, **mib-jnx-ping.txt**, has dependencies on RFC 2925, DiSMAN-PING-MIB, **mib-rfc2925a.txt**. If you try to load **mib-jnx-ping.txt** before loading **mib-rfc2925a.txt**, the compiler returns an error message saying that certain objects in **mib-jnx-ping.txt** are undefined. Load **mib-rfc2925a.txt**, and then try to load **mib-jnx-ping.txt**. The enterprise-specific PING MIB, **mib-jnx-ping.txt**, then loads without any issue.

Related Documentation

- Standard SNMP MIBs Supported by Junos OS on page 13
- Juniper Networks Enterprise-Specific MIBs on page 32

CHAPTER 5

Configuring SNMP

Configuring SNMP on a Device Running Junos OS

Supported Platforms ACX Series, M Series, MX Series, PTX Series, SRX Series, T Series

By default, SNMP is disabled on devices running Junos OS. To enable SNMP on a router or switch, you must include the SNMP configuration statements at the **[edit snmp]** hierarchy level.

To configure the minimum requirements for SNMP, include the following statements at the **[edit snmp]** hierarchy level of the configuration:

```
[edit]
snmp {
  community public;
}
```

The community defined here as **public** grants read access to all MIB data to any client.

To configure complete SNMP features, include the following statements at the **[edit snmp]** hierarchy level:

```
snmp {
  client-list client-list-name {
    ip-addresses;
  }
  community community-name {
    authorization authorization;
    client-list-name client-list-name;
    clients {
      address restrict;
    }
    routing-instance routing-instance-name {
      clients {
      addresses;
      }
    }
    logical-system logical-system-name {
      routing-instance routing-instance-name {
        clients {
        addresses;
        }
      }
    }
    view view-name;
  }
  contact contact;
  description description;
  engine-id {
    (local engine-id | use-mac-address | use-default-ip-address);
  }
  filter-duplicates;
  health-monitor {
    falling-threshold integer;
    interval seconds;
    rising-threshold integer;
  }
  interface [ interface-names ];
  location location;
  name name;
  nonvolatile {
```

```
    commit-delay seconds;
  }
  rmon {
    alarm index {
      description text-description;
      falling-event-index index;
      falling-threshold integer;
      falling-threshold-interval seconds;
      interval seconds;
      request-type (get-next-request | get-request | walk-request);
      rising-event-index index;
      sample-type type;
      startup-alarm alarm;
      syslog-subtag syslog-subtag;
      variable oid-variable;
    }
    event index {
      community community-name;
      description text-description;
      type type;
    }
  }
  traceoptions {
    file filename <files number> <size size> <world-readable | no-world-readable> <match
      regular-expression>;
    flag flag;
  }
  trap-group group-name {
    categories {
      category;
    }
    destination-port port-number;
    routing-instance instance;
    targets {
      address;
    }
    version (all | v1 | v2);
  }
  trap-options {
    agent-address outgoing-interface;
    source-address address;
  }
  view view-name {
    oid object-identifier (include | exclude);
  }
}
```

Related Documentation

- Understanding the SNMP Implementation in Junos OS on page 9
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Complete SNMPv3 Configuration Statements on page 251

Configuring the System Contact on a Device Running Junos OS

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

You can specify an administrative contact for each system being managed by SNMP. This name is placed into the MIB II **sysContact** object. To configure a contact name, include the **contact** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
contact contact;
```

If the name contains spaces, enclose it in quotation marks (" ").

To define a system contact name that contains spaces:

```
[edit]
snmp {
   contact "Juniper Berry, (650) 555-1234";
}
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuring the System Location for a Device Running Junos OS on page 118
- Configuring the System Description on a Device Running Junos OS on page 119
- Configuring the System Name on page 119
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring the System Location for a Device Running Junos OS

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

You can specify the location of each system being managed by SNMP. This string is placed into the MIB II **sysLocation** object. To configure a system location, include the **location** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
location location;
```

If the location contains spaces, enclose it in quotation marks (" ").

To specify the system location:

```
[edit]
snmp {
   location "Row 11, Rack C";
}
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuring the System Contact on a Device Running Junos OS on page 118
- Configuring the System Description on a Device Running Junos OS on page 119
- Configuring the System Name on page 119

- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring the System Description on a Device Running Junos OS

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

You can specify a description for each system being managed by SNMP. This string is placed into the MIB II **sysDescription** object. To configure a description, include the **description** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
description description;
```

If the description contains spaces, enclose it in quotation marks (" ").

To specify the system description:

```
[edit]
snmp {
   description "M40 router with 8 FPCs";
}
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuring the System Contact on a Device Running Junos OS on page 118
- Configuring the System Location for a Device Running Junos OS on page 118
- Configuring the System Name on page 119
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring the System Name

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Junos OS enables you to override the system name by including the **name** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
name name;
```

If the name contains spaces, enclose it in quotation marks (" ").

To specify the system name override:

```
[edit]
snmp {
   name "snmp 1";
}
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuring the System Contact on a Device Running Junos OS on page 118
- Configuring the System Location for a Device Running Junos OS on page 118

- Configuring the System Description on a Device Running Junos OS on page 119
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring the Commit Delay Timer

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

When a router or switch first receives an SNMP nonvolatile **Set** request, a Junos OS XML protocol session opens and prevents other users or applications from changing the candidate configuration (equivalent to the command-line interface [CLI] **configure exclusive** command). If the router does not receive new SNMP **Set** requests within 5 seconds (the default value), the candidate configuration is committed and the Junos OS XML protocol session closes (the configuration lock is released). If the router receives new SNMP **Set** requests while the candidate configuration is being committed, the SNMP **Set** request is rejected and an error is generated. If the router receives new SNMP **Set** requests before 5 seconds have elapsed, the commit-delay timer (the length of time between when the last SNMP request is received and the commit is requested) resets to 5 seconds.

By default, the timer is set to 5 seconds. To configure the timer for the SNMP **Set** reply and start of the commit, include the **commit-delay** statement at the **[edit snmp nonvolatile]** hierarchy level:

```
[edit snmp nonvolatile]
commit-delay seconds;
```

seconds is the length of the time between when the SNMP request is received and the commit is requested for the candidate configuration. For more information about the **configure exclusive** command and locking the configuration, see the *CLI User Guide*.

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring the SNMP Community String

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Configuring the SNMP agent in Junos OS is a straightforward task that shares many familiar settings common to other managed devices in your network. For example, you need to configure Junos OS with an SNMP community string and a destination for traps. Community strings are administrative names that group collections of devices and the agents that are running on them together into common management domains. If a manager and an agent share the same community, they can communicate with each other. An SNMP community defines the level of authorization granted to its members, such as which MIB objects are available, which operations (read-only or read-write) are valid for those objects, and which SNMP clients are authorized, based on their source IP addresses.

The SNMP community string defines the relationship between an SNMP server system and the client systems. This string acts like a password to control the clients' access to the server. To configure a community string in a Junos OS configuration, include the **community** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
community name {
  authorization authorization;
  clients {
    default restrict;
    address restrict;
  }
  view view-name;
}
```

If the community name contains spaces, enclose it in quotation marks (" ").

Community names must be unique.

NOTE: You cannot configure the same community name at the **[edit snmp community]** and **[edit snmp v3 snmp-community *community-index*]** hierarchy levels.

```
[edit groups global]
user@host# set snmp community name
```

This example uses the standard name **public** to create a community that gives limited read-only access.

```
[edit groups global]
user@host# set snmp community public
```

1. Define the authorization level for the community.

   The default authorization level for a community is **read-only**. To allow **Set** requests within a community, you need to define that community as **authorization read-write**. For **Set** requests, you also need to include the specific MIB objects that are accessible with read-write privileges using the **view** statement. The default view includes all supported MIB objects that are accessible with read-only privileges; no MIB objects are accessible with read-write privileges. For more information about the **view** statement, see "Configuring MIB Views" on page 126.

```
[edit groups global snmp community name]
user@host# set authorization authorization
```

This example confines the public community to read-only access. Any SNMP client (for example, an SNMP management system) that belongs to the public community can read MIB variables but cannot set (change) them.

```
[edit groups global snmp community public]
user@host# set authorization read-only
```

2. Define a list of clients in the community who are authorized to communicate with the SNMP agent in Junos OS.

   The **clients** statement lists the IP addresses of the clients (community members) that are allowed to use this community. If no **clients** statement is present, all clients are allowed. For ***address***, you must specify an IPv4 or IPv6 address, not a hostname. Include the **default restrict** option to deny access to all SNMP clients for which access is not explicitly granted. We recommend that you always include the **default restrict** option to limit SNMP client access to the local router.

NOTE: Community names must be unique. You cannot configure the same community name at the **[edit snmp community]** and **[edit snmp v3 snmp-community *community-index*]** hierarchy levels.

Related Documentation

- Adding a Group of Clients to an SNMP Community on page 166
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Examples: Configuring the SNMP Community String on page 122

Examples: Configuring the SNMP Community String

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Grant read-only access to all clients. With the following configuration, the system responds to SNMP **Get**, **GetNext**, and **GetBulk** requests that contain the community string **public**:

```
[edit]
snmp {
  community public {
    authorization read-only;
  }
}
```

Grant all clients read-write access to the ping MIB and **jnxPingMIB**. With the following configuration, the system responds to SNMP **Get**, **GetNext**, **GetBulk**, and **Set** requests that contain the community string **private** and specify an OID contained in the ping MIB or **jnxPingMIB** hierarchy:

```
[edit]
snmp {
  view ping-mib-view {
```

```
        oid pingMIB include;
        oid jnxPingMIB include;
        community private {
          authorization read-write;
          view ping-mib-view;
        }
      }
    }
```

The following configuration allows read-only access to clients with IP addresses in the range **1.2.3.4/24**, and denies access to systems in the range **fe80::1:2:3:4/64**:

```
[edit]
snmp {
  community field-service {
    authorization read-only;
    clients {
      default restrict; # Restrict access to all SNMP clients not explicitly
      # listed on the following lines.
      1.2.3.4/24; # Allow access by all clients in 1.2.3.4/24 except
      fe80::1:2:3:4/64 restrict;# fe80::1:2:3:4/64.
    }
  }
}
```

Related Documentation

- Configuring the SNMP Community String on page 120

Filtering Duplicate SNMP Requests

Supported Platforms LN Series, PTX Series, SRX Series

By default, filtering duplicate **get**, **getNext**, and **getBulk** SNMP requests is disabled on devices running Junos OS. If a network management station retransmits a **Get**, **GetNext**, or **GetBulk** SNMP request too frequently to the router, that request might interfere with the processing of previous requests and slow down the response time of the agent. Filtering these duplicate requests improves the response time of the SNMP agent. Junos OS uses the following information to determine if an SNMP request is a duplicate:

- Source IP address of the SNMP request
- Source UDP port of the SNMP request
- Request ID of the SNMP request

To filter duplicate SNMP requests, include the **filter-duplicates** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
filter-duplicates;
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuring the Interfaces on Which SNMP Requests Can Be Accepted on page 124
- Filtering Interface Information Out of SNMP Get and GetNext Output on page 125

- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring the Interfaces on Which SNMP Requests Can Be Accepted

Supported Platforms M Series, MX Series, PTX Series, QFX Series, T Series

By default, all router or switch interfaces have SNMP access privileges. To limit the access through certain interfaces only, include the **interface** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
interface [ interface-names ];
```

Specify the names of any logical or physical interfaces that should have SNMP access privileges. Any SNMP requests entering the router or switch from interfaces not listed are discarded.

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Example: Configuring Secured Access List Checking on page 124
- *Configuring SNMP*

Example: Configuring Secured Access List Checking

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

SNMP access privileges are granted to only devices on interfaces **so-0/0/0** and **at-1/0/1**. The following example does this by configuring a list of logical interfaces:

```
[edit]
snmp {
   interface [ so-0/0/0.0 so-0/0/0.1 at-1/0/1.0 at-1/0/1.1 ];
}
```

The following example grants the same access by configuring a list of physical interfaces:

```
[edit]
snmp {
   interface [ so-0/0/0 at-1/0/1 ];
}
```

Related Documentation

- Configuring the Interfaces on Which SNMP Requests Can Be Accepted on page 124
- Filtering Interface Information Out of SNMP Get and GetNext Output on page 125
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Filtering Interface Information Out of SNMP Get and GetNext Output

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Junos OS enables you to filter out information related to specific interfaces from the output of SNMP **Get** and **GetNext** requests performed on interface-related MIBs such as IF MIB, ATM MIB, RMON MIB, and the Juniper Networks enterprise-specific IF MIB.

You can use the following options of the **filter-interfaces** statement at the **[edit snmp]** hierarchy level to specify the interfaces that you want to exclude from SNMP **Get** and **GetNext** queries:

- **interfaces**—Interfaces that match the specified regular expressions.
- **all-internal-interfaces**—Internal interfaces.

```
[edit]
  snmp {
  filter-interfaces {
    interfaces {
      interface1;
      interface2;
    }
    all-internal-interfaces;
  }
}
```

Starting with Release 12.1, Junos OS provides an except option (**!** operator) that enables you to filter out all interfaces except those interfaces that match all the regular expressions prefixed with the **!** mark.

For example, to filter out all interfaces except the **ge** interfaces from the SNMP **get** and **get-next** results, enter the following command:

```
[edit snmp]
user@host# set filter-interfaces interfaces "!^~ge-.*"
user@host# commit
```

When this is configured, Junos OS filters out all interfaces except the **ge** interfaces from the SNMP **get** and **get-next** results.

NOTE: The **!** mark is supported only as the first character of the regular expression. If it appears anywhere else in a regular expression, Junos OS considers the regular expression invalid, and returns an error.

However, note that these settings are limited to SNMP operations, and the users can continue to access information related to the interfaces (including those hidden using the **filter-interfaces** options) using the appropriate Junos OS command-line interface (CLI) commands.

Related Documentation

- Configuring the Interfaces on Which SNMP Requests Can Be Accepted on page 124
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring MIB Views

Supported Platforms QFX Series

SNMPv3 defines the concept of MIB views in RFC 3415, *View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*. MIB views provide an agent better control over who can access specific branches and objects within its MIB tree. A view consists of a name and a collection of SNMP object identifiers, which are either explicitly included or excluded. Once defined, a view is then assigned to an SNMPv3 group or SNMPv1/v2c community (or multiple communities), automatically masking which parts of the agent's MIB tree members of the group or community can (or cannot) access.

By default, an SNMP community grants read access and denies write access to all supported MIB objects (even communities configured as **authorization read-write**). To restrict or grant read or write access to a set of MIB objects, you must configure a MIB view and associate the view with a community.

To configure MIB views, include the **view** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
view view-name {
  oid object-identifier (include | exclude);
}
```

The **view** statement defines a MIB view and identifies a group of MIB objects. Each MIB object of a view has a common object identifier (OID) prefix. Each object identifier represents a subtree of the MIB object hierarchy. The subtree can be represented either by a sequence of dotted integers (such as **1.3.6.1.2.1.2**) or by its subtree name (such as **interfaces**). A configuration statement uses a view to specify a group of MIB objects on which to define access. You can also use a wildcard character asterisk (*) to include OIDs that match a particular pattern in the SNMP view. To enable a view, you must associate the view with a community.

NOTE: To remove an OID completely, use the **delete view all oid** *oid-number* command but omit the **include** parameter.

To associate MIB views with a community, include the **view** statement at the **[edit snmp community** *community-name***]** hierarchy level:

```
[edit snmp community community-name]
view view-name;
```

For more information about the Ping MIB, see RFC 2925 and the *PING MIB* topic.

Related Documentation

- *PING MIB*
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Example: Ping Proxy MIB on page 127
- **view (Configuring a MIB View) on page 343**
- *view (Associating MIB View with a Community)*
- **oid on page 293**

Example: Ping Proxy MIB

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Restrict the ***ping-mib*** community to read and write access of the Ping MIB and **jnxpingMIB** only. Read or write access to any other MIB using this community is not allowed.

```
[edit snmp]
view ping-mib-view {
   oid 1.3.6.1.2.1.80 include; #pingMIB
   oid jnxPingMIB include; #jnxPingMIB
}
community ping-mib {
   authorization read-write;
   view ping-mib-view;
}
```

The following configuration prevents the ***no-ping-mib*** community from accessing Ping MIB and **jnxPingMIB** objects. However, this configuration does not prevent the ***no-ping-mib*** community from accessing any other MIB object that is supported on the device.

```
[edit snmp]
view no-ping-mib-view {
   oid 1.3.6.1.2.1.80 exclude; # deny access to pingMIB objects
   oid jnxPingMIB exclude; # deny access to jnxPingMIB objects
}
community no-ping-mib {
   authorization read-write;
   view ping-mib-view;
}
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Configuring MIB Views on page 126
- **view (Configuring a MIB View) on page 343**
- **oid on page 293**

Configuring SNMP Trap Options and Groups on a Device Running Junos OS

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Some carriers have more than one trap receiver that forwards traps to a central NMS. This allows for more than one path for SNMP traps from a router to the central NMS through different trap receivers. A device running Junos OS can be configured to send the same copy of each SNMP trap to every trap receiver configured in the trap group.

The source address in the IP header of each SNMP trap packet is set to the address of the outgoing interface by default. When a trap receiver forwards the packet to the central NMS, the source address is preserved. The central NMS, looking only at the source address of each SNMP trap packet, assumes that each SNMP trap came from a different source.

In reality, the SNMP traps came from the same router, but each left the router through a different outgoing interface.

The statements discussed in the following sections are provided to allow the NMS to recognize the duplicate traps and to distinguish SNMPv1 traps based on the outgoing interface.

To configure SNMP trap options and trap groups, include the **trap-options** and **trap-group** statements at the **[edit snmp]** hierarchy level:

```
[edit snmp]
trap-options {
   agent-address outgoing-interface;
   source-address address;
}
trap-group group-name {
   categories {
      category;
   }
   destination-port port-number;
   targets {
      address;
   }
   version (all | v1 | v2);
}
```

Related Documentation

- Configuring SNMP Trap Options on page 128
- Configuring SNMP Trap Groups on page 132
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring SNMP Trap Options

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Using SNMP trap options, you can set the source address of every SNMP trap packet sent by the router to a single address regardless of the outgoing interface. In addition, you can set the agent address of the SNMPv1 traps. For more information about the contents of SNMPv1 traps, see RFC 1157.

NOTE: SNMP cannot be associated with any routing instances other than the master routing instance.

To configure SNMP trap options, include the **trap-options** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
trap-options {
    agent-address outgoing-interface;
    enterprise-oid
    logical-system
    routing-instance
    source-address address;
}
```

You must also configure a trap group for the trap options to take effect. For information about trap groups, see "Configuring SNMP Trap Groups" on page 132.

This topic contains the following sections:

Configuring the Source Address for SNMP Traps

You can configure the source address of trap packets in many ways: **lo0**, a valid IPv4 address configured on one of the router interfaces, a logical-system address, or the address of a routing-instance. The value **lo0** indicates that the source address of the SNMP trap packets is set to the lowest loopback address configured on the interface **lo0**.

You can configure the source address of trap packets in one of the following formats:

- a valid IPv4 address configured on one of the router interfaces
- **lo0**; that is the lowest loopback address configured on the interface **lo0**.
- a logical-system name
- a routing-instance name

A valid IPv4 Address As the Source Address

To specify a valid interface address as the source address for SNMP traps on one of the router interfaces, include the **source-address** statement at the **[edit snmp trap-options]** hierarchy level:

```
[edit snmp trap-options]
source-address address;
```

address is a valid IPv4 address configured on one of the router interfaces.

The Lowest Loopback Address As the Source Address

To specify the source address of the SNMP traps so that they use the lowest loopback address configured on the interface **lo0** as the source address, include the **source-address** statement at the **[edit snmp trap-options]** hierarchy level:

```
[edit snmp trap-options]
source-address lo0;
```

To enable and configure the loopback address, include the **address** statement at the **[edit interfaces lo0 unit 0 family inet]** hierarchy level:

```
[edit interfaces]
lo0 {
  unit 0 {
    family inet {
      address ip-address;
    }
  }
}
```

To configure the loopback address as the source address of trap packets:

```
[edit snmp]
trap-options {
  source-address lo0;
}
trap-group "urgent-dispatcher" {
  version v2;
  categories link startup;
  targets {
    192.168.10.22;
    172.17.1.2;
  }
}
[edit interfaces]
lo0 {
  unit 0 {
    family inet {
      address 10.0.0.1/32;
      address 127.0.0.1/32;
    }
  }
}
```

In this example, the IP address **10.0.0.1** is the source address of every trap sent from this router.

Logical System Name as the Source Address

To specify a logical system name as the source address of SNMP traps, include the **logical-system *logical-system-name*** statement at the **[edit snmp trap-options]** hierarchy level.

For example, the following configuration sets logical system name **ls1** as the source address of SNMP traps:

```
[edit snmp]
  trap-options{
  logical-system ls1;
```

```
}
```

Routing Instance Name as the Source Address

To specify a routing instance name as the source address of SNMP traps, include the **routing-instance** *routing-instance-name* statement at the **[edit snmp trap-options]** hierarchy level.

For example, the following configuration sets the routing instance name **ri1** as the source address for SNMP traps:

```
[edit snmp]
  trap-options {
  routing-instance ri1;
}
```

Configuring the Agent Address for SNMP Traps

The agent address is only available in SNMPv1 trap packets (see RFC 1157). By default, the router's default local address is used in the agent address field of the SNMPv1 trap. To configure the agent address, include the **agent-address** statement at the **[edit snmp trap-options]** hierarchy level. Currently, the agent address can only be the address of the outgoing interface:

```
[edit snmp]
trap-options {
  agent-address outgoing-interface;
}
```

To configure the outgoing interface as the agent address:

```
[edit snmp]
trap-options {
  agent-address outgoing-interface;
}
trap-group " urgent-dispatcher" {
  version v1;
  categories link startup;
  targets {
    192.168.10.22;
    172.17.1.2;
  }
}
```

In this example, each SNMPv1 trap packet sent has its agent address value set to the IP address of the outgoing interface.

Adding snmpTrapEnterprise Object Identifier to Standard SNMP Traps

The **snmpTrapEnterprise** object helps you identify the enterprise that has defined the trap. Typically, the **snmpTrapEnterprise** object appears as the last varbind in enterprise-specific SNMP version 2 traps. However, starting Release 10.0, Junos OS enables you to add the **snmpTrapEnterprise** object identifier to standard SNMP traps as well.

To add **snmpTrapEnterprise** to standard traps, include the **enterprise-oid** statement at the **[edit snmp trap-options]** hierarchy level. If the **enterprise-oid** statement is not included in the configuration, **snmpTrapEnterprise** is added only for enterprise-specific traps.

```
[edit snmp]
trap-options {
  enterprise-oid;
}
```

Related Documentation

- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Configuring SNMP Trap Groups on page 132
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

Configuring SNMP Trap Groups

Supported Platforms LN Series, SRX Series

You can create and name a group of one or more types of SNMP traps and then define which systems receive the group of SNMP traps. The trap group must be configured for SNMP traps to be sent. To create an SNMP trap group, include the **trap-group** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
trap-group group-name {
  categories {
    category;
  }
  destination-port port-number;
  routing-instance instance;
  targets {
    address;
  }
  version (all | v1 | v2);
}
```

The trap group name can be any string and is embedded in the community name field of the trap. To configure your own trap group port, include the **destination-port** statement. The default destination port is port 162.

For each trap group that you define, you must include the **target** statement to define at least one system as the recipient of the SNMP traps in the trap group. Specify the IPv4 or IPv6 address of each recipient, not its hostname.

Specify the types of traps the trap group can receive in the **categories** statement. For information about the category to which the traps belong, see the "Standard SNMP Traps Supported on Devices Running Junos OS" on page 95 and "Juniper Networks Enterprise-Specific SNMP Traps" on page 80 topics.

Specify the routing instance used by the trap group in the **routing-instance** statement. All targets configured in the trap group use this routing instance.

A trap group can receive the following categories:

- **authentication**—Authentication failures
- **chassis**—Chassis or environment notifications
- **configuration**—Configuration notifications
- **link**—Link-related notifications (up-down transitions, DS-3 and DS-1 line status change, IPv6 interface state change, and Passive Monitoring PIC overload)

NOTE: To send Passive Monitoring PIC overload interface traps, select the link trap category.

- **remote-operations**—Remote operation notifications
- **rmon-alarm**—Alarm for RMON events
- **routing**—Routing protocol notifications
- **sonet-alarms**—SONET/SDH alarms

NOTE: If you omit the SONET/SDH subcategories, all SONET/SDH trap alarm types are included in trap notifications.

  - **loss-of-light**—Loss of light alarm notification
  - **pll-lock**—PLL lock alarm notification
  - **loss-of-frame**—Loss of frame alarm notification
  - **loss-of-signal**—Loss of signal alarm notification
  - **severely-errored-frame**—Severely errored frame alarm notification
  - **line-ais**—Line alarm indication signal (AIS) alarm notification
  - **path-ais**—Path AIS alarm notification
  - **loss-of-pointer**—Loss of pointer alarm notification
  - **ber-defect**—SONET/SDH bit error rate alarm defect notification
  - **ber-fault**—SONET/SDH error rate alarm fault notification
  - **line-remote-defect-indication**—Line remote defect indication alarm notification
  - **path-remote-defect-indication**—Path remote defect indication alarm notification
  - **remote-error-indication**—Remote error indication alarm notification
  - **unequipped**—Unequipped alarm notification
  - **path-mismatch**—Path mismatch alarm notification
  - **loss-of-cell**—Loss of cell delineation alarm notification
  - **vt-ais**—Virtual tributary (VT) AIS alarm notification
  - **vt-loss-of-pointer**—VT loss of pointer alarm notification

  - **vt-remote-defect-indication**—VT remote defect indication alarm notification
  - **vt-unequipped**—VT unequipped alarm notification
  - **vt-label-mismatch**—VT label mismatch error notification
  - **vt-loss-of-cell**—VT loss of cell delineation notification
- **startup**—System warm and cold starts
- **timing-events**—Timing events and defects notification
- **vrrp-events**—Virtual Router Redundancy Protocol (VRRP) events such as new-master or authentication failures
- **startup**—System warm and cold starts
- **vrrp-events**—Virtual Router Redundancy Protocol (VRRP) events such as new-master or authentication failures

If you include SONET/SDH subcategories, only those SONET/SDH trap alarm types are included in trap notifications.

The **version** statement allows you to specify the SNMP version of the traps sent to targets of the trap group. If you specify **v1** only, SNMPv1 traps are sent. If you specify **v2** only, SNMPv2 traps are sent. If you specify **all**, both an SNMPv1 and an SNMPv2 trap are sent for every trap condition. For more information about the **version** statement, see version.

Related Documentation

- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Configuring SNMP Trap Options on page 128
- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Example: Configuring SNMP Trap Groups on page 134

Example: Configuring SNMP Trap Groups

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Set up a trap notification list named **urgent-dispatcher** for link and startup traps. This list is used to identify the network management hosts (**1.2.3.4** and **fe80::1:2:3:4**) to which traps generated by the local router should be sent. The name specified for a trap group is used as the SNMP community string when the agent sends traps to the listed targets.

```
[edit]
snmp {
  trap-group "urgent-dispatcher" {
    version v2;
    categories link startup;
    targets {
      1.2.3.4;
      fe80::1:2:3:4;
    }
  }
```

```
}
```

Related Documentation

- Configuring SNMP Trap Groups on page 132
- Configuring SNMP Trap Options and Groups on a Device Running Junos OS on page 128
- Configuring SNMP Trap Options on page 128

Configuring the Trap Notification Filter

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

SNMPv3 uses the notify filter to define which traps (or which objects from which traps) are sent to the network management system (NMS). The trap notification filter limits the type of traps that are sent to the NMS.

Each object identifier represents a subtree of the MIB object hierarchy. The subtree can be represented either by a sequence of dotted integers (such as **1.3.6.1.2.1.2**) or by its subtree name (such as **interfaces**). You can also use the wildcard character asterisk (*) in the object identifier (OID) to specify object identifiers that match a particular pattern.

To configure the trap notifications filter, include the **notify-filter** statement at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
notify-filter profile-name;
```

profile-name is the name assigned to the notify filter.

By default, the OID is set to **include**. To define access to traps (or objects from traps), include the **oid** statement at the **[edit snmp v3 notify-filter *profile-name*]** hierarchy level:

```
[edit snmp v3 notify-filter profile-name]
oid oid (include | exclude);
```

oid is the object identifier. All MIB objects represented by this statement have the specified OID as a prefix. It can be specified either by a sequence of dotted integers or by a subtree name.

- **include**—Include the subtree of MIB objects represented by the specified OID.
- **exclude**—Exclude the subtree of MIB objects represented by the specified OID.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring the SNMPv3 Trap Notification on page 159
- Configuring the Trap Target Address on page 161
- Defining and Configuring the Trap Target Parameters on page 163
- Configuring SNMP Informs on page 157
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

CHAPTER 6

Configuring SNMPv3

SNMPv3 Overview

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

In contrast to SNMP version 1 (SNMPv1) and SNMP version 2 (SNMPv2), SNMP version 3 (SNMPv3) supports authentication and encryption. SNMPv3 uses the user-based security model (USM) for message security and the view-based access control model (VACM) for access control. USM specifies authentication and encryption. VACM specifies access-control rules.

USM uses the concept of a user for which security parameters (levels of security, authentication, privacy protocols, and keys) are configured for both the agent and the manager. Messages sent using USM are better protected than messages sent with community strings, where passwords are sent in the clear. With USM, messages exchanged between the manager and the agent can have data integrity checking and data origin authentication. USM protects against message delays and message replays by using time indicators and request IDs. Encryption is also available.

To complement the USM, SNMPv3 uses the VACM, a highly granular access-control model for SNMPv3 applications. Based on the concept of applying security policies to the name of the groups querying the agent, the agent decides whether the group is allowed to view or change specific MIB objects. VACM defines collections of data (called views), groups of data users, and access statements that define which views a particular group of users can use for reading, writing, or receiving traps.

Trap entries in SNMPv3 are created by configuring the notify, notify filter, target address, and target parameters. The **notify** statement specifies the type of notification (trap) and contains a single tag. The tag defines a set of target addresses to receive a trap. The notify filter defines access to a collection of trap object identifiers (OIDs). The target address defines a management application's address and other attributes to be used in sending notifications. Target parameters define the message processing and security parameters to be used in sending notifications to a particular management target.

To configure SNMPv3, perform the following tasks:

- Creating SNMPv3 Users on page 138
- Configuring MIB Views on page 126
- Defining Access Privileges for an SNMP Group on page 148
- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring SNMP Informs on page 157

Related Documentation

- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Creating SNMPv3 Users

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

For each SNMPv3 user, you can specify the username, authentication type, authentication password, privacy type, and privacy password. After a user enters a password, a key based on the engine ID and password is generated and is written to the configuration file. After the generation of the key, the password is deleted from this configuration file.

NOTE: You can configure only one encryption type for each SNMPv3 user.

To create users, include the **user** statement at the **[edit snmp v3 usm local-engine]** hierarchy level:

```
[edit snmp v3 usm local-engine]
user username;
```

username is the name that identifies the SNMPv3 user.

To configure user authentication and encryption, include the following statements at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
authentication-md5 {
    authentication-password authentication-password;
}
authentication-sha {
    authentication-password authentication-password;
}
authentication-none;
privacy-aes128 {
    privacy-password privacy-password;
}
privacy-des {
    privacy-password privacy-password;
}
privacy-3des {
    privacy-password privacy-password;
}
privacy-none;
```

Related Documentation

- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Example: Creating SNMPv3 Users Configuration on page 142
- Example: SNMPv3 Configuration on page 139

Example: SNMPv3 Configuration

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Define an SNMPv3 configuration:

```
[edit snmp]
engine-id {
    use-mac-address;
```

```
}
view jnxAlarms {
   oid 1.3.6.1.4.1.2636.3.4 include;
}
view interfaces {
   oid 1.3.6.1.2.1.2 include;
}
view ping-mib {
   oid 1.3.6.1.2.1.80 include;
}
[edit snmp v3]
notify n1 {
   tag router1; # Identifies a set of target addresses
   type trap;# Defines type of notification
}
notify n2 {
   tag host1;
   type trap;
}
notify-filter nf1 {
   oid .1 include; # Defines which traps to send
} # In this case, includes all traps
notify-filter nf2 {
   oid 1.3.6.1.4.1 include; # Sends enterprise-specific traps only
}
notify-filter nf3 {
   oid 1.3.6.1.2.1.1.5 include; # Sends BGP traps only
}
snmp-community index1 {
   community-name "$9$JOZi.QF/AtOz3"; # SECRET-DATA
   security-name john; # Matches the security name at the target parameters
   tag host1; # Finds the addresses that are allowed to be used with
}
target-address ta1 {# Associates the target address with the group
            # san-francisco.
   address 10.1.1.1;
   address-mask 255.255.255.0; # Defines the range of addresses
   port 162;
   tag-list router1;
   target-parameters tp1; # Applies configured target parameters
}
target-address ta2 {
   address 10.1.1.2;
   address-mask 255.255.255.0;
   port 162;
   tag-list host1;
   target-parameters tp2;
}
target-address ta3 {
   address 10.1.1.3;
   address-mask 255.255.255.0;
   port 162;
   tag-list “router1 host1”;
   target-parameters tp3;
}
target-parameters tp1 { # Defines the target parameters
```

```
    notify-filter nf1; # Specifies which notify filter to apply
    parameters {
      message-processing-model v1;
      security-model v1;
      security-level none;
      security-name john; # Matches the security name configured at the
    } # [edit snmp v3 snmp-community community-index hierarchy level.
  }
  target-parameters tp2 {
    notify-filter nf2;
    parameters {
      message-processing-model v1;
      security-model v1;
      security-level none;
      security-name john;
    }
  }
  target-parameters tp3 {
    notify-filter nf3;
    parameters {
      message-processing-model v1;
      security-model v1;
      security-level none;
      security-name john;
    }
  }
  usm {
    local-engine { #Defines authentication and encryption for SNMPv3 users
      user user1 {
        authentication-md5 {
          authentication-password authentication-password;
        }
        privacy-des {
          privacy-password privacy-password;
        }
      }
      user user2 {
        authentication-sha {
          authentication-password authentication-password;
        }
        privacy-none;
      }
      user user3 {
        authentication-none;
        privacy-none;
      }
      user user4 {
        authentication-sha {
          authentication-password authentication-password;
        }
        privacy-aes128 {
          privacy-password privacy-password;
        }
      }
      user user5 {
        authentication-sha {
```

```
            authentication-password authentication-password;
          }
          privacy-none;
        }
      }
    }
    vacm {
      access {
        group san-francisco { #Defines the access privileges for the group
          default-context-prefix { # called san-francisco
            security-model v1 {
              security-level none {
                notify-view ping-mib;
                read-view interfaces;
                write-view jnxAlarms;
              }
            }
          }
        }
      }
      security-to-group {
        security-model v1 {
          security-name john { # Assigns john to the security group
            group san-francisco; # called san-francisco
          }
          security-name bob {
            group new-york;
          }
          security-name elizabeth {
            group chicago;
          }
        }
      }
    }
```

Related Documentation

- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Example: Creating SNMPv3 Users Configuration

Supported Platforms SRX Series

Define SNMPv3 users:

```
[edit]
snmp {
  v3 {
    usm {
      local-engine {
        user user1 {
          authentication-md5 {
            authentication-password authentication-password;
          }
          privacy-des {
```

```
            privacy-password password;
          }
        }
        user user2 {
          authentication-sha {
            authentication-password authentication-password;
          }
          privacy-none;
        }
        user user3 {
          authentication-none;
          privacy-none;
        }
        user user4 {
          authentication-md5 {
            authentication-password authentication-password;
          }
          privacy-des {
            privacy-password authentication-password;
          }
        }
        user user5 {
          authentication-sha {
            authentication-password authentication-password;
          }
          privacy-aes128 {
            privacy-password authentication-password;
          }
        }
      }
    }
  }
}
```

Related Documentation

- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Minimum SNMPv3 Configuration on a Device Running Junos OS

Supported Platforms M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

To configure the minimum requirements for SNMPv3, include the following statements at the **[edit snmp v3]** and **[edit snmp]** hierarchy levels:

NOTE: You must configure at least one view (notify, read, or write) at the **[edit snmp view-name]** hierarchy level.

```
[edit snmp]
view view-name {
  oid object-identifier (include | exclude);
}
[edit snmp v3]
```

```
notify name {
  tag tag-name;
}
notify-filter profile-name {
  oid object-identifier (include | exclude);
}
snmp-community community-index {
  security-name security-name;
}
target-address target-address-name {
  address address;
  target-parameters target-parameters-name;
}
target-parameters target-parameters-name {
  notify-filter profile-name;
  parameters {
    message-processing-model (v1 | v2c | v3);
    security-level (authentication | none | privacy);
    security-model (usm | v1 | v2c);
    security-name security-name;
  }
}
usm {
  local-engine {
    user username {
    }
  }
}
vacm {
  access {
    group group-name {
      (default-context-prefix | context-prefix context-prefix){
        security-model (any | usm | v1 | v2c) {
          security-level (authentication | none | privacy) {
            notify-view view-name;
            read-view view-name;
            write-view view-name;
          }
        }
      }
    }
  }
  security-to-group {
    security-model (usm | v1 | v2c) {
      security-name security-name {
        group group-name;
      }
    }
  }
}
```

Related Documentation

- Creating SNMPv3 Users on page 138
- Configuring MIB Views on page 126
- Defining Access Privileges for an SNMP Group on page 148

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring SNMP Informs on page 157
- Complete SNMPv3 Configuration Statements on page 251
- Example: SNMPv3 Configuration on page 139

Configuring the SNMPv3 Authentication Type

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

By default, in a Junos OS configuration the SNMPv3 authentication type is set to none.

This topic includes the following sections:

- Configuring MD5 Authentication on page 145
- Configuring SHA Authentication on page 145
- Configuring No Authentication on page 146

Configuring MD5 Authentication

To configure the message digest algorithm (MD5) as the authentication type for an SNMPv3 user, include the **authentication-md5** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
authentication-md5 {
  authentication-password authentication-password;
}
```

authentication-password is the password used to generate the key used for authentication.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring SHA Authentication

To configure the secure hash algorithm (SHA) as the authentication type for an SNMPv3 user, include the **authentication-sha** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
authentication-sha {
  authentication-password authentication-password;
}
```

authentication-password is the password used to generate the key used for authentication.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring No Authentication

To configure no authentication for an SNMPv3 user, include the **authentication-none** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
authentication-none;
```

Related Documentation

- Configuring the Encryption Type on page 146
- Defining Access Privileges for an SNMP Group on page 148
- Configuring the Access Privileges Granted to a Group on page 149
- Assigning Security Model and Security Name to a Group on page 153
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring the Encryption Type

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

By default, encryption is set to none.

NOTE: Before you configure encryption, you must configure MD5 or SHA authentication.

Before you configure the **privacy-des**, **privacy-3des** and **privacy-aes128** statements, you must install the **jcrypto** package, and either restart the SNMP process or reboot the router.

This topic includes the following sections:

- Configuring the Advanced Encryption Standard Algorithm on page 147
- Configuring the Data Encryption Algorithm on page 147
- Configuring Triple DES on page 147
- Configuring No Encryption on page 148

Configuring the Advanced Encryption Standard Algorithm

To configure the Advanced Encryption Standard (AES) algorithm for an SNMPv3 user, include the **privacy-aes128** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
privacy-aes128 {
    privacy-password privacy-password;
}
```

privacy-password is the password used to generate the key used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring the Data Encryption Algorithm

To configure the data encryption algorithm (DES) for an SNMPv3 user, include the **privacy-des** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
privacy-des {
    privacy-password privacy-password;
}
```

privacy-password is the password used to generate the key used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring Triple DES

To configure triple DES for an SNMPv3 user, include the **privacy-3des** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
privacy-3des {
    privacy-password privacy-password;
}
```

privacy-password is the password used to generate the key used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Configuring No Encryption

To configure no encryption for an SNMPv3 user, include the **privacy-none** statement at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level:

```
[edit snmp v3 usm local-engine user username]
privacy-none;
```

Related Documentation

- Configuring the SNMPv3 Authentication Type on page 145
- Defining Access Privileges for an SNMP Group on page 148
- Configuring the Access Privileges Granted to a Group on page 149
- Assigning Security Model and Security Name to a Group on page 153
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Defining Access Privileges for an SNMP Group

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

The SNMP version 3 (SNMPv3) uses the view-based access control model (VACM), which allows you to configure the access privileges granted to a group. Access is controlled by filtering the MIB objects available for a specific operation through a predefined view. You assign views to determine the objects that are visible for read, write, and notify operations for a particular group, using a particular context, a particular security model (v1, v2c, or usm), and particular security level (authenticated, privacy, or none). For information about how to configure views, see "Configuring MIB Views" on page 126.

You define user access to management information at the **[edit snmp v3 vacm]** hierarchy level. All access control within VACM operates on groups, which are collections of users as defined by USM, or community strings as defined in the SNMPv1 and SNMPv2c security models. The term ***security-name*** refers to these generic end users. The group to which a specific security name belongs is configured at the **[edit snmp v3 vacm security-to-group]** hierarchy level. That security name can be associated with a group defined at the **[edit snmp v3 vacm security-to-group]** hierarchy level. A group identifies a collection of SNMP users that share the same access policy. You then define the access privileges associated with a group at the **[edit snmp v3 vacm access]** hierarchy level. Access privileges are defined using views. For each group, you can apply different views depending on the SNMP operation; for example, read (**get**, **getNext**, or **getBulk**) write (**set**), notifications, the security level used (authentication, privacy, or none), and the security model (v1, v2c, or usm) used within an SNMP request.

You configure members of a group with the **security-name** statement. For v3 packets using USM, the security name is the same as the username. For SNMPv1 or SNMPv2c

packets, the security name is determined based on the community string. Security names are specific to a security model. If you are also configuring VACM access policies for SNMPv1 or SNMPv2c packets, you must assign security names to groups for each security model (SNMPv1 or SNMPv2c) at the **[edit snmp v3 vacm security-to-group]** hierarchy level. You must also associate a security name with an SNMP community at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level.

To configure the access privileges for an SNMP group, include statements at the **[edit snmp v3 vacm]** hierarchy level:

```
[edit snmp v3 vacm]
access {
  group group-name {
    (default-context-prefix | context-prefix context-prefix){
      security-model (any | usm | v1 | v2c) {
        security-level (authentication | none | privacy) {
          notify-view view-name;
          read-view view-name;
          write-view view-name;
        }
      }
    }
  }
}
security-to-group {
  security-model (usm | v1 | v2c) {
    security-name security-name {
      group group-name;
    }
  }
}
```

Related Documentation

- Configuring the SNMPv3 Authentication Type on page 145
- Configuring the Access Privileges Granted to a Group on page 149
- Assigning Security Model and Security Name to a Group on page 153
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring the Access Privileges Granted to a Group

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

This topic includes the following sections:

- Configuring the Group on page 150
- Configuring the Security Model on page 150
- Configuring the Security Level on page 150
- Associating MIB Views with an SNMP User Group on page 151

Configuring the Group

To configure the access privileges granted to a group, include the **group** statement at the **[edit snmp v3 vacm access]** hierarchy level:

```
[edit snmp v3 vacm access]
group group-name;
```

group-name is a collection of SNMP users that belong to a common SNMP list that defines an access policy. Users belonging to a particular SNMP group inherit all access privileges granted to that group.

Configuring the Security Model

To configure the security model, include the **security-model** statement at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*)]** hierarchy level:

```
[edit snmp v3 vacm access group group-name (default-context-prefix | context-prefix
    context-prefix)]
security-model (any | usm | v1 | v2c);
```

- **any**—Any security model
- **usm**—SNMPv3 security model
- **v1**—SNMPV1 security model
- **v2c**—SNMPv2c security model

Configuring the Security Level

To configure the access privileges granted to packets with a particular security level, include the **security-level** statement at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c)]** hierarchy level:

```
[edit snmp v3 vacm access group group-name default-context-prefix security-model (any
    | usm | v1 | v2c)]
security-level (authentication | none | privacy);
```

- **none**—Provides no authentication and no encryption.
- **authentication**—Provides authentication but no encryption.
- **privacy**—Provides authentication and encryption.

NOTE: Access privileges are granted to all packets with a security level equal to or greater than that configured. If you are configuring the SNMPv1 or SNMPv2c security model, use **none** as your security level. If you are configuring the SNMPv3 security model (USM), use the **authentication**, **none**, or **privacy** security level.

Associating MIB Views with an SNMP User Group

MIB views define access privileges for members of a group. Separate views can be applied for each SNMP operation (read, write, and notify) within each security model (usm, v1, and v2c) and each security level (authentication, none, and privacy) supported by SNMP.

To associate MIB views with an SNMP user group, include the following statements at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]** hierarchy level:

```
[edit snmp v3 vacm accessgroup group-name (default-context-prefix | context-prefix
   context-prefix)security-model (any | usm | v1 | v2c) security-level (authentication | none
   | privacy)]
notify-view view-name;
read-view view-name;
write-view view-name;
```

NOTE: You must associate at least one view (notify, read, or write) at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]** hierarchy level.

You must configure the MIB view at the **[edit snmp view *view-name*]** hierarchy level. For information about how to configure MIB views, see "Configuring MIB Views" on page 126.

This section describes the following topics related to this configuration:

Configuring the Notify View

To associate notify access with an SNMP user group, include the **notify-view** statement at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]** hierarchy level:

```
[edit snmp v3 vacm access group group-name (default-context-prefix | context-prefix
   context-prefix) security-model (any | usm | v1 | v2c) security-level (authentication | none
   | privacy)]
notify-view view-name;
```

view-name specifies the notify access, which is a list of notifications that can be sent to each user in an SNMP group. A view name cannot exceed 32 characters.

Configuring the Read View

To associate a read view with an SNMP group, include the **read-view** statement at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]** hierarchy level:

```
[edit snmp v3 vacm access group group-name (default-context-prefix | context-prefix
  context-prefix) security-model (any | usm | v1 | v2c) security-level (authentication | none
  | privacy)]
read-view view-name;
```

view-name specifies read access for an SNMP user group. A view name cannot exceed 32 characters.

Configuring the Write View

To associate a write view with an SNMP user group, include the **write-view** statement at the **[edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]** hierarchy level:

```
[edit snmp v3 vacm access group group-name (default-context-prefix | context-prefix
  context-prefix) security-model (any | usm | v1 | v2c) security-level (authentication | none
  | privacy)]
write-view view-name;
```

view-name specifies write access for an SNMP user group. A view name cannot exceed 32 characters.

Related Documentation

- Configuring the SNMPv3 Authentication Type on page 145
- Defining Access Privileges for an SNMP Group on page 148
- Assigning Security Model and Security Name to a Group on page 153
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Example: Access Privilege Configuration on page 152

Example: Access Privilege Configuration

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Define access privileges:

```
[edit snmp v3]
access {
  group group1 {
    default-context-prefix {
      security-model usm {        #Define an SNMPv3 security model
        security-level privacy {
          notify-view nv1;
          read-view rv1;
          write-view wv1;
```

```
        }
      }
    }
    context-prefix lr1/ri1{ # routing instance ri1 in logical system lr1
      security-model usm {
        security-level privacy {
          notify-view nv1;
          read-view rv1;
          write-view wv1;
        }
      }
    }
  }
  group group2 {
    default-context-prefix {
      security-model usm {        #Define an SNMPv3 security model
        security-level authentication {
          read-view rv2;
          write-view wv2;
        }
      }
    }
  }
  group group3 {
    default-context-prefix {
      security-model v1 {        #Define an SNMPv3 security model
        security-level none {
          read-view rv3;
          write-view wv3;
        }
      }
    }
  }
}
```

Related Documentation

- Configuring the Access Privileges Granted to a Group on page 149
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Assigning Security Model and Security Name to a Group

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

To assign security names to groups, include the following statements at the **[edit snmp v3 vacm security-to-group]** hierarchy level:

```
[edit snmp v3 vacm security-to-group]
security-model (usm | v1 | v2c) {
  security-name security-name {
    group group-name;
  }
}
```

This topic includes the following sections:

Configuring the Security Model

To configure the security model, include the **security-model** statement at the **[edit snmp v3 vacm security-to-group]** hierarchy level:

```
[edit snmp v3 vacm security-to-group]
security-model (usm | v1 | v2c);
```

- **usm**—SNMPv3 security model
- **v1**—SNMPv1 security model
- **v2c**—SNMPv2 security model

Assigning Security Names to Groups

To associate a security name with an SNMPv3 user, or a v1 or v2 community string, include the **security-name** statement at the **[edit snmp v3 vacm security-to-group security-model (usm | v1 | v2c)]** hierarchy level:

```
[edit snmp v3 vacm security-to-group security-model (usm | v1 | v2c)]
security-name security-name;
```

For SNMPv3, the ***security-name*** is the username configured at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level. For SNMPv1 and SNMPv2c, the security name is the community string configured at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level. For information about configuring usernames, see "Creating SNMPv3 Users" on page 138. For information about configuring a community string, see "Configuring the SNMPv3 Community" on page 167.

NOTE: The USM security name is separate from the SNMPv1 and SNMPv2c security name. If you support SNMPv1 and SNMPv2c in addition to SNMPv3, you must configure separate security names within the security-to-group configuration at the **[edit snmp v3 vacm access]** hierarchy level.

Configuring the Group

After you have created SNMPv3 users, or v1 or v2 security names, you associate them with a group. A group is a set of security names belonging to a particular security model. A group defines the access rights for all users belonging to it. Access rights define what SNMP objects can be read, written to, or created. A group also defines what notifications a user is allowed to receive.

If you already have a group that is configured with all of the view and access permissions that you want to give a user, you can add the user to that group. If you want to give a user

view and access permissions that no other groups have, or if you do not have any groups configured, create a group and add the user to it.

To configure the access privileges granted to a group, include the **group** statement at the **[edit snmp v3 vacm security-to-group security-model (usm | v1 | v2c) security-name *security-name*]** hierarchy level:

```
[edit snmp v3 vacm security-to-group security-model (usm | v1 | v2c) security-name
  security-name]
group group-name;
```

group-name identifies a collection of SNMP security names that share the same access policy. For more information about groups, see "Defining Access Privileges for an SNMP Group" on page 148.

Example: Security Group Configuration

Supported Platforms LN Series, M Series, MX Series, SRX Series, T Series

Assign security names to groups:

```
vacm {
  security-to-group {
    security-model usm {
      security-name user1 {
        group group1;
      }
      security-name user2 {
        group group2;
      }
      security-name user3 {
        group group3;
      }
    }
  }
}
```

Related Documentation

- Assigning Security Model and Security Name to a Group on page 153
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Example: Configuring the Tag List

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

In the following example, two tag entries (**router1** and **router2**) are defined at the **[edit snmp v3 notify *notify-name*]** hierarchy level. When an event triggers a notification, Junos OS sends a trap to all target addresses that have **router1** or **router2** configured in their target-address tag list. This results in the first two targets getting one trap each, and the third target getting two traps.

```
[edit snmp v3]
notify n1 {
  tag router1; # Identifies a set of target addresses
```

```
    type trap; # Defines the type of notification
  }
  notify n2 {
    tag router2;
    type trap;
  }
  target-address ta1 {
    address 10.1.1.1;
    address-mask 255.255.255.0;
    port 162;
    tag-list router1;
    target-parameters tp1;
  }
  target-address ta2 {
    address 10.1.1.2;
    address-mask 255.255.255.0;
    port 162;
    tag-list router2;
    target-parameters tp2;
  }
  target-address ta3 {
    address 10.1.1.3;
    address-mask 255.255.255.0;
    port 162;
    tag-list “router1 router2”; #Define multiple tags in the target address tag list
    target-parameters tp3;
  }
```

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring the Trap Target Address on page 161
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring the Local Engine ID

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

By default, the local engine ID uses the default IP address of the router. The local engine ID is the administratively unique identifier for the SNMPv3 engine. This statement is optional. To configure the local engine ID, include the **engine-id** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
engine-id {
  (local engine-id-suffix | use-default-ip-address | use-mac-address);
}
```

- **local *engine-id-suffix***—The engine ID suffix is explicitly configured.
- **use-default-ip-address**—The engine ID suffix is generated from the default IP address.
- **use-mac-address**—The SNMP engine identifier is generated from the Media Access Control (MAC) address of the management interface on the router.

The local engine ID is defined as the administratively unique identifier of an SNMPv3 engine, and is used for identification, not for addressing. There are two parts of an engine ID: prefix and suffix. The prefix is formatted according to the specifications defined in RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*. You can configure the suffix here.

NOTE: SNMPv3 authentication and encryption keys are generated based on the associated passwords and the engine ID. If you configure or change the engine ID, you must commit the new engine ID before you configure SNMPv3 users. Otherwise the keys generated from the configured passwords are based on the previous engine ID. For the engine ID, we recommend using the master IP address of the device if the device has multiple routing engines and has the master IP address configured. Alternatively, you can use the MAC address of the management port if the device has only one Routing Engine.

Related Documentation

- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Example: SNMPv3 Configuration on page 139

Configuring SNMP Informs

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

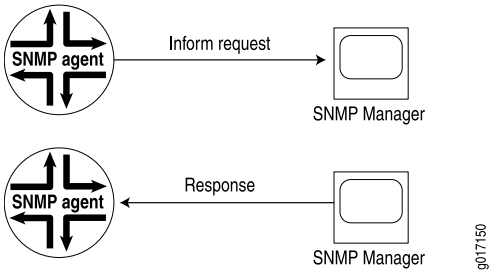
Junos OS supports two types of notifications: traps and informs. With traps, the receiver does not send any acknowledgment when it receives a trap. Therefore, the sender cannot determine if the trap was received. A trap may be lost because a problem occurred during transmission. To increase reliability, an inform is similar to a trap except that the inform is stored and retransmitted at regular intervals until one of these conditions occurs:

- The receiver (target) of the inform returns an acknowledgment to the SNMP agent.
- A specified number of unsuccessful retransmissions have been attempted and the agent discards the inform message.

If the sender never receives a response, the inform can be sent again. Thus, informs are more likely to reach their intended destination than traps are. Informs use the same communications channel as traps (same socket and port) but have different protocol data unit (PDU) types.

Informs are more reliable than traps, but they consume more network, router, and switch resources (see Figure 1 on page 158). Unlike a trap, an inform is held in memory until a response is received or the timeout is reached. Also, traps are sent only once, whereas an inform may be retried several times. Use informs when it is important that the SNMP manager receive all notifications. However, if you are more concerned about network traffic, or router and switch memory, use traps.

Figure 1: Inform Request and Response



For information about configuring SNMP traps, see "Configuring SNMPv3 Traps on a Device Running Junos OS" on page 158.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring the Remote Engine and Remote User on page 172
- Configuring the Inform Notification Type and Target Address on page 170
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring SNMPv3 Traps on a Device Running Junos OS

Supported Platforms ACX Series, M Series, MX Series, PTX Series, QFX Series, T Series

In SNMPv3, you create traps and informs by configuring the **notify**, **target-address**, and **target-parameters** parameters. Traps are unconfirmed notifications, whereas informs are confirmed notifications. This section describes how to configure SNMP traps. For information about configuring SNMP informs, see "Configuring SNMP Informs" on page 157.

The target address defines a management application's address and parameters to be used in sending notifications. Target parameters define the message processing and security parameters that are used in sending notifications to a particular management target. SNMPv3 also lets you define SNMPv1 and SNMPv2c traps.

NOTE: When you configure SNMP traps, make sure your configured access privileges allow the traps to be sent. Access privileges are configured at the **[edit snmp v3 vacm access]** and **[edit snmp v3 vacm security-to-group]** hierarchy levels.

To configure SNMP traps, include the following statements at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
notify name {
    tag tag-name;
    type trap;
}
notify-filter name {
```

```
    oid object-identifier (include | exclude);
  }
  target-address target-address-name {
    address address;
    address-mask address-mask;
    logical-system logical-system;
    port port-number;
    routing-instance instance;
    tag-list tag-list;
    target-parameters target-parameters-name;
  }
  target-parameters target-parameters-name {
    notify-filter profile-name;
    parameters {
      message-processing-model (v1 | v2c | v3);
      security-level (authentication | none | privacy);
      security-model (usm | v1 | v2c);
      security-name security-name;
    }
  }
```

Related Documentation

- Configuring the SNMPv3 Trap Notification on page 159
- Configuring the Trap Notification Filter on page 135
- Configuring the Trap Target Address on page 161
- Defining and Configuring the Trap Target Parameters on page 163
- Configuring SNMP Informs on page 157
- Configuring the Remote Engine and Remote User on page 172
- Configuring the Inform Notification Type and Target Address on page 170

Configuring the SNMPv3 Trap Notification

Supported Platforms M Series, MX Series, PTX Series, QFX Series, T Series

The **notify** statement specifies the type of notification (trap) and contains a single tag. The tag defines a set of target addresses to receive a trap. The tag list contains one or more tags and is configured at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level. If the tag list contains this tag, Junos OS sends a notification to all the target addresses associated with this tag.

To configure the trap notifications, include the **notify** statement at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
notify name {
  tag tag-name;
  type trap;
}
```

name is the name assigned to the notification.

tag-name defines the target addresses to which this notification is sent. This notification is sent to all the target-addresses that have this tag in their tag list. The ***tag-name*** is not included in the notification.

trap is the type of notification.

NOTE: Each notify entry name must be unique.

Junos OS supports two types of notification: trap and inform.

For information about how to configure the tag list, see "Configuring the Trap Target Address" on page 162.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring the Trap Notification Filter on page 135
- Configuring the Trap Target Address on page 161
- Defining and Configuring the Trap Target Parameters on page 163
- Configuring SNMP Informs on page 157
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Example: Configuring SNMPv3 Trap Notification

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

Specify three sets of destinations to send traps:

```
[edit snmp v3]
notify n1 {
   tag router1;
   type trap;
}
notify n2 {
   tag router2;
   type trap
}
notify n3 {
   tag router3;
   type trap;
}
```

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring the Trap Target Address

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

The target address defines a management application's address and parameters that are used in sending notifications. It can also identify management stations that are allowed to use specific community strings. When you receive a packet with a recognized community string and a tag is associated with it, Junos OS looks up all the target addresses with this tag and verifies that the source address of this packet matches one of the configured target addresses.

NOTE: You must configure the address mask when you configure the SNMP community.

To specify where you want the traps to be sent and define what SNMPv1 and SNMPv2cc packets are allowed, include the **target-address** statement at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
target-address target-address-name;
```

target-address-name is the string that identifies the target address.

To configure the target address properties, include the following statements at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
address address;
address-mask address-mask;
logical-system logical-system;
port port-number;
routing-instance instance;
tag-list tag-list;
target-parameters target-parameters-name;
```

This section includes the following topics:

Configuring the Address

To configure the address, include the **address** statement at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
address address;
```

address is the SNMP target address.

Configuring the Address Mask

The address mask specifies a set of addresses that are allowed to use a community string and verifies the source addresses for a group of target addresses.

To configure the address mask, include the **address-mask** statement at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
address-mask address-mask;
```

address-mask combined with the address defines a range of addresses. For information about how to configure the community string, see "Configuring the SNMPv3 Community" on page 167.

Configuring the Port

By default, the UDP port is set to 162. To configure a different port number, include the **port** statement at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
port port-number;
```

port-number is the SNMP target port number.

Configuring the Routing Instance

Traps are sent over the default routing instance. To configure the routing instance for sending traps, include the **routing-instance** statement at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
routing-instance instance;
```

instance is the name of the routing instance. To configure a routing instance within a logical system, specify the logical system name followed by the routing instance name. Use a slash (/) to separate the two names (for example, **test-lr/test-ri**). To configure the default routing instance on a logical system, specify the logical system name followed by **default** (for example, **test-lr/default**).

Configuring the Trap Target Address

Each **target-address** statement can have one or more tags configured in its tag list. Each tag can appear in more than one tag list. When a significant event occurs on the network device, the tag list identifies the targets to which a notification is sent.

To configure the tag list, include the **tag-list** statement at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
tag-list "tag-list";
```

tag-list specifies one or more tags as a space-separated list enclosed within double quotes.

For an example of tag list configuration, see "Example: Configuring the Tag List" on page 155.

For information about how to specify a tag at the **[edit snmp v3 notify *notify-name*]** hierarchy level, see "Configuring the SNMPv3 Trap Notification" on page 159.

NOTE: When you configure SNMP traps, make sure your configured access privileges allow the traps to be sent. Configure access privileges at the **[edit snmp v3 vacm access]** hierarchy level.

Applying Target Parameters

The **target-parameters** statement at the **[edit snmp v3]** hierarchy level applies the target parameters configured at the **[edit snmp v3 target-parameters *target-parameters-name*]** hierarchy level.

To reference configured target parameters, include the **target-parameters** statement at the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
[edit snmp v3 target-address target-address-name]
target-parameters target-parameters-name;
```

target-parameters-name is the name associated with the message processing and security parameters that are used in sending notifications to a particular management target.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring the SNMPv3 Trap Notification on page 159
- Configuring the Trap Notification Filter on page 135
- Defining and Configuring the Trap Target Parameters on page 163
- Configuring SNMP Informs on page 157
- Complete SNMPv3 Configuration Statements on page 251

Defining and Configuring the Trap Target Parameters

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Target parameters define the message processing and security parameters that are used in sending notifications to a particular management target.

To define a set of target parameters, include the **target-parameters** statement at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
target-parameters target-parameters-name;
```

target-parameters-name is the name assigned to the target parameters.

To configure target parameter properties, include the following statements at the **[edit snmp v3 target-parameters *target-parameter-name*]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name]
notify-filter profile-name;
parameters {
  message-processing-model (v1 | v2c | V3);
  security-level (authentication | none | privacy);
  security-model (usm | v1 | v2c);
  security-name security-name;
}
```

This topic includes the following sections:

Applying the Trap Notification Filter

To apply the trap notification filter, include the **notify-filter** statement at the **[edit snmp v3 target-parameters *target-parameter-name*]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name]
notify-filter profile-name;
```

profile-name is the name of a configured notify filter. For information about configuring notify filters, see "Configuring the Trap Notification Filter" on page 135.

Configuring the Target Parameters

To configure target parameter properties, include the following statements at the **[edit snmp v3 target-parameters *target-parameter-name* parameters]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
message-processing-model (v1 | v2c | v3);
security-level (authentication | none | privacy);
security-model (usm | v1 | v2c);
security-name security-name;
```

This section includes the following topics:

Configuring the Message Processing Model

The message processing model defines which version of SNMP to use when generating SNMP notifications. To configure the message processing model, include the **message-processing-model** statement at the **[edit snmp v3 target-parameters *target-parameter-name* parameters]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
message-processing-model (v1 | v2c | v3);
```

- **v1**—SNMPv1 message processing model
- **v2c**—SNMPv2c message processing model
- **v3**—SNMPV3 message processing model

Configuring the Security Model

To define the security model to use when generating SNMP notifications, include the **security-model** statement at the **[edit snmp v3 target-parameters *target-parameter-name* parameters]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
security-model (usm | v1 | v2c);
```

- **usm**—SNMPv3 security model
- **v1**—SNMPv1 security model
- **v2c**—SNMPv2c security model

Configuring the Security Level

The **security-level** statement specifies whether the trap is authenticated and encrypted before it is sent.

To configure the security level to use when generating SNMP notifications, include the **security-level** statement at the **[edit snmp v3 target-parameters *target-parameter-name* parameters]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
security-level (authentication | none | privacy);
```

- **authentication**—Provides authentication but no encryption.
- **none**—No security. Provides no authentication and no encryption.
- **privacy**—Provides authentication and encryption.

NOTE: If you are configuring the SNMPv1 or SNMPV2c security model, use **none** as your security level. If you are configuring the SNMPv3 (USM) security model, use the **authentication** or **privacy** security level.

Configuring the Security Name

To configure the security name to use when generating SNMP notifications, include the **security-name** statement at the **[edit snmp v3 target-parameters *target-parameter-name* parameters]** hierarchy level:

```
[edit snmp v3 target-parameters target-parameter-name parameters]
security-name security-name;
```

If the USM security model is used, the **security-name** identifies the user that is used when the notification is generated. If the v1 or v2c security models are used, **security-name** identifies the SNMP community used when the notification is generated.

NOTE: The access privileges for the group associated with a security name must allow this notification to be sent.

If you are using the v1 or v2 security models, the security name at the **[edit snmp v3 vacm security-to-group]** hierarchy level must match the security name at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring the SNMPv3 Trap Notification on page 159
- Configuring the Trap Notification Filter on page 135
- Configuring the Trap Target Address on page 161
- Configuring SNMP Informs on page 157
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Adding a Group of Clients to an SNMP Community

Supported Platforms ACX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Junos OS enables you to add one or more groups of clients to an SNMP community. You can include the **client-list-name *name*** statement at the **[edit snmp community *community-name*]** hierarchy level to add all the members of the client list or prefix list to an SNMP community.

To define a list of clients, include the **client-list** statement followed by the IP addresses of the clients at the **[edit snmp]** hierarchy level:

```
[edit snmp]
  client-list client-list-name {
  ip-addresses;
}
```

You can configure a prefix list at the **[edit policy options]** hierarchy level. Support for prefix lists in the SNMP community configuration enables you to use a single list to configure the SNMP and routing policies. For more information about the **prefix-list** statement, see the *Routing Policy Configuration Guide*.

To add a client list or prefix list to an SNMP community, include the **client-list-name** statement at the **[edit snmp community *community-name*]** hierarchy level:

```
[edit snmp community community-name]
client-list-name client-list-name;
```

NOTE: The client list and prefix list must not have the same name.

The following example shows how to define a client list:

```
[edit]
snmp {
   client-list clentlist1 {
      10.1.1.1/32;
      10.2.2.2/32;
   }
}
```

The following example shows how to add a client list to an SNMP community:

```
[edit]
snmp {
   community community1 {
      authorization read-only;
      client-list-name clientlist1;
   }
}
```

The following example shows how to add a prefix list to an SNMP community:

```
[edit]
policy-options {
   prefix-list prefixlist {
   10.3.3.3/32;
   10.5.5.5/32;
   }
}
snmp {
   community community2 {
   client-list-name prefixlist;
   }
}
```

Related Documentation

- *client-list*
- *client-list-name*

Configuring the SNMPv3 Community

Supported Platforms ACX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

The SNMP community defines the relationship between an SNMP server system and the client systems. This statement is optional.

To configure the SNMP community, include the **snmp-community** statement at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
snmp-community community-index;
```

community-index is the index for the SNMP community.

To configure the SNMP community properties, include the following statements at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level:

```
[edit snmp v3 snmp-community community-index]
community-namecommunity-name;
contextcontext-name;
security-name security-name;
tag tag-name;
```

This section includes the following topics:

Configuring the Community Name

The community name defines the SNMP community. The SNMP community authorizes SNMPv1 or SNMPv2c clients. The access privileges associated with the configured security name define which MIB objects are available and the operations (read, write, or notify) allowed on those objects.

To configure the SNMP community name, include the **community-name** statement at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level:

```
[edit snmp v3 snmp-community community-index]
community-namecommunity-name;
```

community-name is the community string for an SNMPv1 or SNMPv2c community.

If unconfigured, it is the same as the community index.

If the community name contains spaces, enclose it in quotation marks (“ “).

NOTE: Community names must be unique. You cannot configure the same community name at the **[edit snmp community]** and **[edit snmp v3 snmp-community *community-index*]** hierarchy levels. The configured community name at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level is encrypted. You cannot view the community name after you have configured it and committed your changes. In the command-line interface (CLI), the community name is concealed.

Configuring the Context

An SNMP context defines a collection of management information that is accessible to an SNMP entity. Typically, an SNMP entity has access to multiple contexts. A context can be a physical or logical system, a collection of multiple systems, or even a subset of a system. Each context in a management domain has a unique identifier.

To configure an SNMP context, include the **context** *context-name* statement at the **[edit snmp v3 snmp-community** *community-index***]** hierarchy level:

```
[edit snmp v3 snmp-community community-index]
context context-name;
```

NOTE: To query a routing instance or a logical system,

Configuring the Security Names

To assign a community string to a security name, include the **security-name** statement at the **[edit snmp v3 snmp-community** *community-index***]** hierarchy level:

```
[edit snmp v3 snmp-community community-index]
security-name security-name;
```

security-name is used when access control is set up. The **security-to-group** configuration at the **[edit snmp v3 vacm]** hierarchy level identifies the group.

NOTE: This security name must match the security name configured at the **[edit snmp v3 target-parameters** *target-parameters-name* **parameters]** hierarchy level when you configure traps.

Configuring the Tag

To configure the tag, include the **tag** statement at the **[edit snmp v3 snmp-community** *community-index***]** hierarchy level:

```
[edit snmp v3 snmp-community community-index]
tagtag-name;
```

tag-name identifies the address of managers that are allowed to use a community string.

Related Documentation

- Creating SNMPv3 Users on page 138
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Example: SNMPv3 Community Configuration on page 169

Example: SNMPv3 Community Configuration

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Define an SNMP community:

```
[edit snmp v3]
snmp-community index1 {
   community-name "$9$JOZi.QF/AtOz3"; # SECRET-DATA
   security-name john;
   tag router1; # Identifies managers that are allowed to use
```

```
    # a community string
    target-address ta1 {
      address 10.1.1.1;
      address-mask 255.255.255.0; # Defines the range of addresses
      port 162;
      tag-list router1;
      target-parameters tp1; # Applies configured target parameters
    }
  }
```

Related Documentation

- Configuring the SNMPv3 Community on page 167
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring the Inform Notification Type and Target Address

Supported Platforms ACX Series, M Series, MX Series, PTX Series, T Series

To configure the inform notification type and target information, include the following statements at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
notify name {
  tag tag-name;
  type (trap | inform);
}
target-address target-address-name {
  address address;
  address-mask address-mask;
  logical-system logical-system;
  port port-number;
  retry-count number;
  routing-instance instance;
  tag-list tag-list;
  target-parameters target-parameters-name;
  timeout seconds;
}
target-parameters target-parameters-name {
  notify-filter profile-name;
  parameters {
    message-processing-model (v1 | v2c | v3);
    security-level (authentication | none | privacy);
    security-model (usm | v1 | v2c);
    security-name security-name;
  }
}
```

notify *name* is the name assigned to the notification. Each notify entry name must be unique.

tag *tag-name* defines the target addresses that are sent this notification. The notification is sent to all target addresses that have this tag in their tag list. The ***tag-name*** is not

included in the notification. For information about how to configure the tag list, see "Configuring the Trap Target Address" on page 162.

type inform is the type of notification.

target-address ***target-address-name*** identifies the target address. The target address defines a management application's address and parameters that are used to respond to informs.

timeout ***seconds*** is the number of seconds to wait for an acknowledgment. If no acknowledgment is received within the timeout period, the inform is retransmitted. The default timeout is **15** seconds.

retry-count ***number*** is the maximum number of times an inform is transmitted if no acknowledgment is received. The default is **3**. If no acknowledgment is received after the inform is transmitted the maximum number of times, the inform message is discarded.

message-processing-model defines which version of SNMP to use when SNMP notifications are generated. Informs require a **v3** message processing model.

security-model defines the security model to use when SNMP notifications are generated. Informs require a **usm** security model.

security-model defines the security model to use when SNMP notifications are generated. Informs require a **usm** security model.

security-level specifies whether the inform is authenticated and encrypted before it is sent. For the **usm** security model, the security level must be one of the following:

- **authentication**—Provides authentication but no encryption.
- **privacy**—Provides authentication and encryption.

security-name identifies the username that is used when generating the inform.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring SNMP Informs on page 157
- Configuring the Remote Engine and Remote User on page 172
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Example: Configuring the Inform Notification Type and Target Address on page 171

Example: Configuring the Inform Notification Type and Target Address

Supported Platforms ACX Series, M Series, MX Series, PTX Series, SRX Series, T Series

In the following example, target **172.17.20.184** is configured to respond to informs. The inform timeout is **30** seconds and the maximum retransmit count is **3**. The inform is sent to all targets in the **tl1** list. The security model for the remote user is **usm** and the remote engine username is **u10**.

```
[edit snmp v3]
notify n1 {
    type inform;
    tag tl1;
}
notify-filter nf1 {
    oid .1.3 include;
}
target-address ta1 {
    address 172.17.20.184;
    retry-count 3;
    tag-list tl1;
    address-mask 255.255.255.0;
    target-parameters tp1;
    timeout 30;
}
target-parameters tp1 {
    parameters {
        message-processing-model v3;
        security-model usm;
        security-level privacy;
        security-name u10;
    }
    notify-filter nf1;
}
```

Related Documentation

- Configuring the Inform Notification Type and Target Address on page 170
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

Configuring the Remote Engine and Remote User

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

To send inform messages to an SNMPv3 user on a remote device, you must first specify the engine identifier for the SNMP agent on the remote device where the user resides. The remote engine ID is used to compute the security digest for authenticating and encrypting packets sent to a user on the remote host. When sending an inform message, the agent uses the credentials of the user configured on the remote engine (inform target).

To configure a remote engine and remote user to receive and respond to SNMP informs, include the following statements at the **[edit snmp v3]** hierarchy level:

```
[edit snmp v3]
usm {
    remote-engine engine-id {
        user username {
            authentication-md5 {
```

```
            authentication-key key;
          }
          authentication-none;
          authentication-sha {
            authentication-key key;
          }
          privacy-3des {
            privacy-key key;
          }
          privacy-aes128 {
            privacy-key key;
          }
          privacy-des {
            privacy-key key;
          }
          privacy-none;
        }
      }
    }
```

For informs, **remote-engine *engine-id*** is the identifier for the SNMP agent on the remote device where the user resides.

For informs, **user *username*** is the user on a remote SNMP engine who receives the informs.

Informs generated can be **unauthenticated**, **authenticated**, or **authenticated_and_encrypted**, depending on the security level of the SNMPv3 user configured on the remote engine (the inform receiver). The authentication key is used for generating message authentication code (MAC). The privacy key is used to encrypt the inform PDU part of the message.

Related Documentation

- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring SNMP Informs on page 157
- Configuring the Inform Notification Type and Target Address on page 170
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Example: Configuring the Remote Engine ID and Remote Users on page 173

Example: Configuring the Remote Engine ID and Remote Users

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

This example shows how to configure a remote engine and remote user so you can receive and respond to SNMP inform notifications. Inform notifications can be authenticated and encrypted. They are also more reliable than traps, another type of notification that Junos OS supports. Unlike traps, inform notifications are stored and retransmitted at regular intervals until one of these conditions occurs:

- The target of the inform notification returns an acknowledgment to the SNMP agent.

- A specified number of unsuccessful retransmissions have been attempted.

Requirements

No special configuration beyond device initialization is required before configuring this example.

This feature requires the use of plain-text passwords valid for SNMPv3. SNMPv3 has the following special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Although quotation marks are not always required to enclose passwords, it is best to use them. You need quotation marks if the password contains any spaces or possibly in the case of certain special characters or punctuation.

Overview

Inform notifications are supported in SNMPv3 to increase reliability. For example, an SNMP agent receiving an inform notification acknowledges the receipt.

For inform notifications, the remote engine ID identifies the SNMP agent on the remote device where the user resides, and the username identifies the user on a remote SNMP engine who receives the inform notifications.

Consider a scenario in which you have the values in Table 17 on page 174 to use in configuring the remote engine ID and remote user in this example.

Table 17: Values to Use in Example

Name of Variable	Value
username	u10
remote engine ID	800007E5804089071BC6D10A41
authentication type	authentication-md5
authentication password	qol67R%?
encryption type	privacy-des
privacy password	m*72Jl9v

Configuration

CLI Quick Configuration

To quickly configure this example, copy the following commands and paste them into a text file, remove any line breaks and change any details necessary to match your network configuration, copy and paste these commands into the CLI at the **[edit snmp v3]** hierarchy level, and then enter **commit** from configuration mode.

The following example configures user **u10** located on remote engine **0x800007E5804089071BC6D10A41** and the user's authentication and privacy keys. The keys are autogenerated from the passwords entered by the command-line interface (CLI) user.

[Warning: element unresolved in stylesheets: <step> (in <example>). This is probably a new element that is not yet supported in the stylesheets.]

Configure the remote engine ID, username, and authentication type and password.

```
[edit snmp v3]
user@host# set usm remote-engine 800007E5804089071BC6D10A41 user u10
  authentication-md5 authentication-key "qol67R%?"
```

[Warning: element unresolved in stylesheets: <step> (in <example>). This is probably a new element that is not yet supported in the stylesheets.]

Configure the encryption type and privacy password.

You can configure only one encryption type per SNMPv3 user.

```
[edit snmp v3]
user@host# set usm remote-engine 800007E5804089071BC6D10A41 user u10
  privacy-des privacy-key "m*72Jl9v"
```

[Warning: element unresolved in stylesheets: <results> (in <example>). This is probably a new element that is not yet supported in the stylesheets.]

In configuration mode, confirm your configuration by entering the **show** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit snmp v3]
user@ host# show
usm {
  remote-engine 800007E5804089071BC6D10A41 {
    user u10 {
      authentication-md5 {
        authentication-key "$9$D0jP536901Riktu1IcSwY2gUj5QF3
        /CYgQF/Cu0xN-bwgZGiqP5iH.5TF/9WLX7wYoaUkqfoaAp
        0BEhSreW87s24aUjsY4ZDjq.RhcyWLNdbg4Zs
        YJDHkTQ69Apu1EcyrvWQF/tuOREYg4ajHmPQF39
        Ygz3n6At8XxNYgik.PTz7-ikmfn6vW8XVw";
      }
      privacy-des {
        privacy-key "$9$/gyNCu1KvWdwYMWw2gJHkRhcrWx"; ## SECRET-DATA
      }
      privacy-des {
        privacy-key "$9$MZZXxdwYgJUjlKJGiH5T69Au0IrlM7NbeK24
        aJDjO1IRylM8Xbwg1R24aJDjHqm5n/Ap0ORhn6evLXbwmf5T
        /CRhSyKM5QEcleW87-Vbs4JGD.mT-VwgaZkqfTznAphSrlM8yr
        Wx7dsYTzF36AtuO1EcpuNdwYoa69CuRhcyleM8rlaZGjq.O1IEhr";
      }
```

```
    }
  }
```

After you have confirmed that the configuration is correct, enter **commit** from configuration mode.

Verification

Verifying the Configuration of the Remote Engine ID and Username

Purpose Verify the status of the engine ID and user information.

Action Display information about the SNMPv3 engine ID and user.

```
user@host> show snmp v3
Local engine ID: 80 00 0a 4c 01 0a ff 03 e3
Engine boots:           3
Engine time:       769187 seconds
Max msg size:       65507 bytes

Engine ID: 80 00 07 e5 80 40 89 07 1b c6 d1 0a 41
    User                            Auth/Priv   Storage      Status
    u10                              md5/des    nonvolatile  active
```

Meaning The output displays the following information:

- Local engine ID and detail about the engine
- Remote engine ID (labeled **Engine ID**)
- Username
- Authentication type and encryption (privacy) type that is configured for the user
- Type of storage for the username, either nonvolatile (configuration saved) or volatile (not saved)
- Status of the new user; only users with an active status can use SNMPv3

Related Documentation

- Configuring the Remote Engine and Remote User on page 172
- Complete SNMPv3 Configuration Statements on page 251
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143
- Configuring SNMP Informs on page 157
- Configuring the Remote Engine and Remote User on page 172

CHAPTER 7

Configuring Routing Instances

Understanding SNMP Support for Routing Instances

Supported Platforms LN Series, M Series, MX Series, SRX Series, T Series

Junos OS enables SNMP managers for all routing instances to request and manage SNMP data related to the corresponding routing instances and logical system networks.

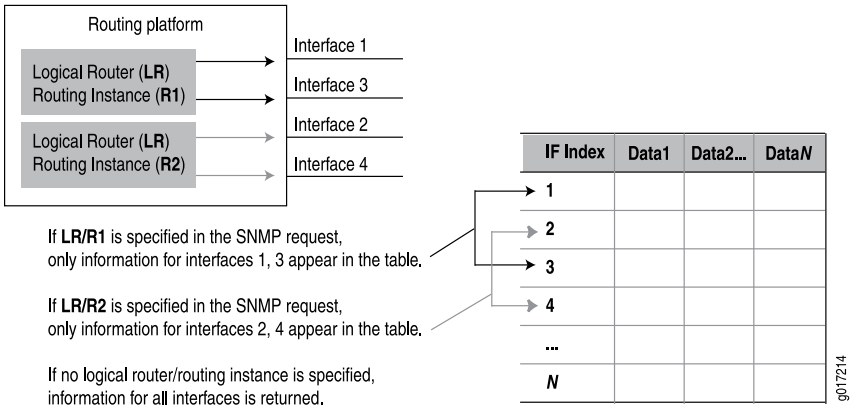
In Junos OS:

- Clients from routing instances other than the default can access MIB objects and perform SNMP operations only on the logical system networks to which they belong.
- Clients from the default routing instance can access information related to all routing instances and logical system networks.

Before Junos OS Release 8.4, only the SNMP manager in the default routing instance (**inet.0**) had access to the MIB objects

With the increase in virtual private network (VPN) service offerings, this feature is useful particularly for service providers who need to obtain SNMP data for specific routing instances (see Figure 2 on page 178). Service providers can use this information for their own management needs or export the data for use by their customers.

Figure 2: SNMP Data for Routing Instances



If no routing instance is specified in the request, the SNMP agent operates as before:

- For nonrouting table objects, all instances are exposed.
- For routing table objects, only those associated with the default routing instance are exposed.

NOTE: The actual protocol data units (PDUs) are still exchanged over the default (inet.0) routing instance, but the data contents returned are dictated by the routing instance specified in the request PDUs.

Related Documentation

- Trap Support for Routing Instances on page 178
- Identifying a Routing Instance on page 179
- Enabling SNMP Access over Routing Instances on page 180
- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180
- Configuring Access Lists for SNMP Access over Routing Instances on page 183

Trap Support for Routing Instances

Supported Platforms LN Series, M Series, MX Series, SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, SRX5800, vSRX, T Series

You can restrict the trap receivers from receiving traps that are not related to the logical system networks to which they belong. To do this, include the **logical-system-trap-filter** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
logical-system-trap-filter;
```

If the **logical-system-trap-filter** statement is not included in the SNMP configuration, all traps are forwarded to the configured routing instance destinations. However, even when this statement is configured, the trap receiver associated with the default routing instance will receive all SNMP traps.

When configured under the trap-group object, all v1 and v2c traps that apply to routing instances (or interfaces belonging to a routing instance) have the routing instance name encoded in the community string. The encoding is identical to that used in request PDUs.

For traps configured under the v3 framework, the routing instance name is carried in the context field when the v3 message processing model has been configured. For other message processing models (v1 or v2c), the routing instance name is not carried in the trap message header (and not encoded in the community string).

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- MIB Support Details on page 63

Identifying a Routing Instance

Supported Platforms ACX Series, M Series, MX Series, PTX Series, SRX Series, T Series

With this feature, routing instances are identified by either the context field in v3 requests or encoded in the community string in v1 or v2c requests.

When encoded in a community string, the routing instance name appears first and is separated from the actual community string by the @ character.

To avoid conflicts with valid community strings that contain the @ character, the community is parsed only if typical community string processing fails. For example, if a routing instance named **RI** is configured, an SNMP request with **RI@public** is processed within the context of the **RI** routing instance. Access control (views, source address restrictions, access privileges, and so on) is applied according to the actual community string (the set of data after the @ character—in this case **public**). However, if the community string **RI@public** is configured, the protocol data unit (PDU) is processed according to that community and the embedded routing instance name is ignored.

Logical systems perform a subset of the actions of a physical router and have their own unique routing tables, interfaces, policies, and routing instances. When a routing instance is defined within a logical system, the logical system name must be encoded along with the routing instance using a slash (**/**) to separate the two. For example, if the routing instance **RI** is configured within the logical system **LS**, that routing instance must be encoded within a community string as **LS/RI@public**. When a routing instance is configured outside a logical system (within the default logical system), no logical system name (or **/** character) is needed.

Also, when a logical system is created, a default routing instance (named **default**) is always created within the logical system. This name should be used when querying data for that routing instance (for example, **LS/default@public**). For v3 requests, the name ***logical system/routing instance*** should be identified directly in the context field.

NOTE: To identify a virtual LAN (VLAN) spanning-tree instance (VSTP on MX Series 3D Universal Edge Routers), specify the routing instance name followed by a double colon (::) and the VLAN ID. For example, to identify VSTP instance for VLAN 10 in the global default routing instance, include **default::10@public** in the **context** (SNMPv3) or **community** (SNMPv1 or v2) string.

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- Enabling SNMP Access over Routing Instances on page 180
- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180

Enabling SNMP Access over Routing Instances

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

To enable SNMP managers in routing instances other than the default routing instance to access SNMP information, include the **routing-instance-access** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
routing-instance-access;
```

If this statement is not included in the SNMP configuration, SNMP managers from routing instances other than the default routing instance cannot access SNMP information.

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- Identifying a Routing Instance on page 179
- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180
- Configuring Access Lists for SNMP Access over Routing Instances on page 183

Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

You can specify the routing instance along with the client information when you add a client to an SNMP community. To specify the routing instance to which a client belongs, include the **routing-instance** statement followed by the routing instance name and client information in the SNMP configuration.

The following example shows the configuration statement to add routing instance **test-ri** to SNMP community **community1**.

NOTE: Routing instances specified at the **[edit snmp community *community-name*]** hierarchy level are added to the default logical system in the community.

```
[edit snmp]
community community1 {
  clients {
    10.209.152.33/32;
  }
  routing-instance test-ri {
    clients {
      10.19.19.1/32;
    }
  }
}
```

If the routing instance is defined within a logical system, include the **routing-instance** statement at the [**edit snmp community *community-name* logical-system *logical-system-name***] hierarchy level, as in the following example:

```
[edit snmp]
community community1 {
  clients {
    10.209.152.33/32;
  }
  logical-system test-LS {
    routing-instance test-ri {
      clients {
        10.19.19.1/32;
      }
    }
  }
}
```

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- Identifying a Routing Instance on page 179
- Enabling SNMP Access over Routing Instances on page 180
- Configuring Access Lists for SNMP Access over Routing Instances on page 183
- Example: Configuring Interface Settings for a Routing Instance on page 181

Example: Configuring Interface Settings for a Routing Instance

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

This example shows an **802.3ad ae0** interface configuration allocated to a routing instance named **INFrtd**:

```
[edit chassis]
aggregated-devices {
  ethernet {
    device-count 5;
  }
}
[edit interfaces ae0]
vlan-tagging;
aggregated-ether-options {
  minimum-links 2;
```

```
    link-speed 100m;
}
unit 0 {
    vlan-id 100;
    family inet {
        address 10.1.0.1/24;
    }
}
[edit interfaces fe-1/1/0]
fastether-options {
    802.3ad ae0;
}
[edit interfaces fe-1/1/1]
fastether-options {
    802.3ad ae0;
}
[edit routing-instances]
INFrtd {
    instance-type virtual-router;
    interface fe-1/1/0.0;
    interface fe-1/1/1.0;
    interface fe-1/1/5.0;
    interface ae0.0;
    protocols {
        ospf {
            area 0.0.0.0 {
                interface all;
            }
        }
    }
}
```

The following **snmpwalk** command shows how to retrieve SNMP-related information from **router1** and the 802.3ae bundle interface belonging to routing instance **INFrtd** with the SNMP community **public**:

```
router# snmpwalk -Os router1 INFrtd@public dot3adAggTable
dot3adAggMACAddress.59 = 0:90:69:92:93:f0
dot3adAggMACAddress.65 = 0:90:69:92:93:f0
dot3adAggActorSystemPriority.59 = 0
dot3adAggActorSystemPriority.65 = 0
dot3adAggActorSystemID.59 = 0:0:0:0:0:0
dot3adAggActorSystemID.65 = 0:0:0:0:0:0
dot3adAggAggregateOrIndividual.59 = true(1)
dot3adAggAggregateOrIndividual.65 = true(1)
dot3adAggActorAdminKey.59 = 0
dot3adAggActorAdminKey.65 = 0
dot3adAggActorOperKey.59 = 0
dot3adAggActorOperKey.65 = 0
dot3adAggPartnerSystemID.59 = 0:0:0:0:0:0
dot3adAggPartnerSystemID.65 = 0:0:0:0:0:0
dot3adAggPartnerSystemPriority.59 = 0
dot3adAggPartnerSystemPriority.65 = 0
dot3adAggPartnerOperKey.59 = 0
dot3adAggPartnerOperKey.65 = 0
dot3adAggCollectorMaxDelay.59 = 0
```

```
dot3adAggCollectorMaxDelay.65 = 0
```

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180

Configuring Access Lists for SNMP Access over Routing Instances

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

You can create and maintain access lists to manage access to SNMP information. Access list configuration enables you to allow or deny SNMP access to clients of a specific routing instance.

The following example shows how to create an access list:

```
[edit snmp]
routing-instance-access {
   access-list {
      ri1 restrict;
      ls1/default;
      ls1/ri2;
      ls1*;
   }
}
```

The configuration given in the example:

- Restricts clients in **ri1** from accessing SNMP information.
- Allows clients in **ls1/default**, **ls1/ri2**, and all other routing instances with names starting with **ls1** to access SNMP information.

You can use the wildcard character (*) to represent a string in the routing instance name.

NOTE: You cannot restrict the SNMP manager of the default routing instance from accessing SNMP information.

Related Documentation

- Understanding SNMP Support for Routing Instances on page 177
- Enabling SNMP Access over Routing Instances on page 180
- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180

CHAPTER 8

Configuring Remote Operations

SNMP Remote Operations Overview

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

A SNMP remote operation is any process on the router that can be controlled remotely using SNMP. Junos OS currently provides support for two SNMP remote operations: the Ping MIB and Traceroute MIB, defined in RFC 2925. Using these MIBs, an SNMP client in the network management system (NMS) can:

- Start a series of operations on a router
- Receive notification when the operations are complete
- Gather the results of each operation

Junos OS also provides extended functionality to these MIBs in the Juniper Networks enterprise-specific extensions **jnxPingMIB** and **jnxTraceRouteMIB**. For more information about **jnxPingMIB** and **jnxTraceRouteMIB**, see *PING MIB* and *Traceroute MIB*.

This topic covers the following sections:

SNMP Remote Operation Requirements

To use SNMP remote operations, you should be experienced with SNMP conventions. You must also configure Junos OS to allow the use of the remote operation MIBs.

Setting SNMP Views

All remote operation MIBs supported by Junos OS require that the SNMP clients have read-write privileges. The default SNMP configuration of Junos OS does not provide clients with a community string with such privileges.

To set read-write privileges for an SNMP community string, include the following statements at the **[edit snmp]** hierarchy level:

```
[edit snmp]
community community-name {
  authorization authorization;
  view view-name;
}
view view-name {
  oid object-identifier (include | exclude);
}
```

Example: Setting SNMP Views

To create a community named **remote-community** that grants SNMP clients read-write access to the Ping MIB, **jnxPing** MIB, Traceroute MIB, and **jnxTraceRoute** MIB, include the following statements at the **[edit snmp]** hierarchy level:

```
snmp {
  view remote-view {
    oid 1.3.6.1.2.1.80 include; # pingMIB
    oid 1.3.6.1.4.1.2636.3.7 include; # jnxPingMIB
    oid 1.3.6.1.2.1.81 include; # traceRouteMIB
    oid 1.3.6.1.4.1.2636.3.8 include; # jnxTraceRouteMIB
  }
  community remote-community {
    view remote-view;
    authorization read-write;
  }
}
```

For more information about the **community** statement, see "Configuring the SNMP Community String" on page 120 and **community**.

For more information about the **view** statement, see "Configuring MIB Views" on page 126, **view (Associating a MIB View with a Community)**, and **view (Configuring a MIB View)**.

Setting Trap Notification for Remote Operations

In addition to configuring the remote operations MIB for trap notification, you must also configure Junos OS. You must specify a target host for remote operations traps.

To configure trap notification for SNMP remote operations, include the **categories** and **targets** statements at the **[edit snmp trap-group *group-name*]** hierarchy level:

```
[edit snmp trap-group group-name]
  categories {
    category;
  }
  targets {
    address;
  }
}
```

Example: Setting Trap Notification for Remote Operations

Specify **172.17.12.213** as a target host for all remote operation traps:

```
snmp {
  trap-group remote-traps {
    categories remote-operations;
    targets {
      172.17.12.213;
    }
  }
}
```

For more information about trap groups, see "Configuring SNMP Trap Groups" on page 132.

Using Variable-Length String Indexes

All tabular objects in the remote operations MIBs supported by Junos OS are indexed by two variables of type **SnmpAdminString**. For more information about **SnmpAdminString**, see RFC 2571.

Junos OS does not handle **SnmpAdminString** any differently from the octet string variable type. However, the indexes are defined as variable length. When a variable length string is used as an index, the length of the string must be included as part of the object identifier (OID).

Example: Set Variable-Length String Indexes

To reference the **pingCtlTargetAddress** variable of a row in **pingCtlTable** where **pingCtlOwnerIndex** is **bob** and **pingCtlTestName** is **test**, use the following object identifier (OID):

```
pingMIB.pingObjects.pingCtlTable.pingCtlEntry.pingCtlTargetAddress."bob"."test"
1.3.6.1.2.1.80.1.2.1.4.3.98.111.98.4.116.101.115.116
```

For more information about the definition of the Ping MIB, see RFC 2925.

Enabling Logging

The SNMP error code returned in response to SNMP requests can only provide a generic description of the problem. The error descriptions logged by the remote operations process can often provide more detailed information about the problem and help you to solve the problem faster. This logging is not enabled by default. To enable logging, include the **flag general** statement at the **[edit snmp traceoptions]** hierarchy level:

```
[edit]
snmp {
  traceoptions {
    flag general;
  }
}
```

For more information about traceoptions, see "Tracing SNMP Activity on a Device Running Junos OS" on page 197.

If the remote operations process receives an SNMP request that it cannot accommodate, the error is logged in the **/var/log/rmopd** file. To monitor this log file, issue the **monitor start rmopd** command in operational mode of the command-line interface (CLI).

Related Documentation

- Using the Ping MIB for Remote Monitoring Devices Running Junos OS on page 188
- Using the Traceroute MIB for Remote Monitoring Devices Running Junos OS on page 195

Using the Ping MIB for Remote Monitoring Devices Running Junos OS

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

A ping test is used to determine whether packets sent from the local host reach the designated host and are returned. If the designated host can be reached, the ping test provides the approximate round-trip time for the packets. Ping test results are stored in **pingResultsTable** and **pingProbeHistoryTable**.

RFC 2925 is the authoritative description of the Ping MIB in detail and provides the ASN.1 MIB definition of the Piing MIB.

Related Documentation

- SNMP Remote Operations Overview on page 185
- Starting a Ping Test on page 188
- Monitoring a Running Ping Test on page 190
- Gathering Ping Test Results on page 192
- Stopping a Ping Test on page 194
- Interpreting Ping Variables on page 194

Starting a Ping Test

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Before you start a ping test, configure a Ping MIB view. This allows SNMP **Set** requests on **pingMIB**. To start a ping test, create a row in **pingCtlTable** and set **pingCtlAdminStatus** to **enabled**. The minimum information that must be specified before setting **pingCtlAdminStatus** to **enabled** is:

- **pingCtlOwnerIndexSnmpAdminString**
- **pingCtlTestNameSnmpAdminString**
- **pingCtlTargetAddressInetAddress**
- **pingCtlTargetAddressTypeInetAddressType**
- **pingCtlRowStatusRowStatus**

For all other values, defaults are chosen unless otherwise specified. **pingCtlOwnerIndex** and **pingCtlTestName** are used as the index, so their values are specified as part of the object identifier (OID). To create a row, set **pingCtlRowStatus** to **createAndWait** or **createAndGo** on a row that does not already exist. A value of **active** for **pingCtlRowStatus** indicates that all necessary information has been supplied and the test can begin; **pingCtlAdminStatus** can be set to **enabled**. An SNMP **Set** request that sets **pingCtlRowStatus** to **active** will fail if the necessary information in the row is not specified or is inconsistent. For information about how to configure a view, see "Setting SNMP Views" on page 186.

There are two ways to start a ping test:

- Using Multiple Set Protocol Data Units (PDUs) on page 189
- Using a Single Set PDU on page 189

Using Multiple Set Protocol Data Units (PDUs)

You can use multiple **Set** request PDUs (multiple PDUs, with one or more varbinds each) and set the following variables in this order to start the test:

- **pingCtlRowStatus** to **createAndWait**
- All appropriate test variables
- **pingCtlRowStatus** to **active**

  Junos OS now verifies that all necessary information to run a test has been specified.
- **pingCtlAdminStatus** to **enabled**

Using a Single Set PDU

You can use a single **Set** request PDU (one PDU, with multiple varbinds) to set the following variables to start the test:

- **pingCtlRowStatus** to **createAndGo**
- All appropriate test variables
- **pingCtlAdminStatus** to **enabled**

Monitoring a Running Ping Test

Supported Platforms LN Series, SRX Series

When **pingCtlAdminStatus** is successfully set to **enabled**, the following is done before the acknowledgment of the SNMP **Set** request is sent back to the client:

- **pingResultsEntry** is created if it does not already exist.
- **pingResultsOperStatus** transitions to **enabled**.

For more information, see the following sections:

- pingResultsTable on page 190
- pingProbeHistoryTable on page 191
- Generating Traps on page 192

pingResultsTable

While the test is running, **pingResultsEntry** keeps track of the status of the test. The value of **pingResultsOperStatus** is **enabled** while the test is running and **disabled** when it has stopped.

The value of **pingCtlAdminStatus** remains **enabled** until you set it to **disabled**. Thus, to get the status of the test, you must examine **pingResultsOperStatus**.

The **pingCtlFrequency** variable can be used to schedule many tests for one **pingCtlEntry**. After a test ends normally (you did not stop the test) and the **pingCtlFrequency** number of seconds has elapsed, the test is started again just as if you had set **pingCtlAdminStatus** to **enabled**. If you intervene at any time between repeated tests (you set **pingCtlAdminStatus** to **disabled** or **pingCtlRowStatus** to **notInService**), the repeat feature is disabled until another test is started and ends normally. A value of 0 for **pingCtlFrequency** indicates this repeat feature is not active.

pingResultsIpTgtAddr and **pingResultsIpTgtAddrType** are set to the value of the resolved destination address when the value of **pingCtlTargetAddressType** is **dns**. When a test starts successfully and **pingResultsOperStatus** transitions to **enabled**:

- **pingResultsIpTgtAddr** is set to **null-string**.
- **pingResultsIpTgtAddrType** is set to **unknown**.

pingResultsIpTgtAddr and **pingResultsIpTgtAddrType** are not set until **pingCtlTargetAddress** can be resolved to a numeric address. To retrieve these values, poll **pingResultsIpTgtAddrType** for any value other than **unknown** after successfully setting **pingCtlAdminStatus** to **enabled**.

At the start of a test, **pingResultsSentProbes** is initialized to 1 and the first probe is sent. **pingResultsSentProbes** increases by 1 each time a probe is sent.

As the test runs, every **pingCtlTimeOut** seconds, the following occur:

- **pingProbeHistoryStatus** for the corresponding **pingProbeHistoryEntry** in **pingProbeHistoryTable** is set to **requestTimedOut**.
- A **pingProbeFailed** trap is generated, if necessary.
- An attempt is made to send the next probe.

NOTE: **No more than one outstanding probe exists for each test.**

For every probe, you can receive one of the following results:

- The target host acknowledges the probe with a response.
- The probe times out; there is no response from the target host acknowledging the probe.
- The probe could not be sent.

Each probe result is recorded in **pingProbeHistoryTable**. For more information about **pingProbeHistoryTable**, see "pingProbeHistoryTable" on page 191.

When a response is received from the target host acknowledging the current probe:

- **pingResultsProbeResponses** increases by 1.
- The following variables are updated:
  - **pingResultsMinRtt**—Minimum round-trip time
  - **pingResultsMaxRtt**—Maximum round-trip time
  - **pingResultsAverageRtt**—Average round-trip time
  - **pingResultsRttSumOfSquares**—Sum of squares of round-trip times
  - **pingResultsLastGoodProbe**—Timestamp of the last response

NOTE: **Only probes that result in a response from the target host contribute to the calculation of the round-trip time (RTT) variables.**

When a response to the last probe is received or the last probe has timed out, the test is complete.

pingProbeHistoryTable

An entry in **pingProbeHistoryTable** (**pingProbeHistoryEntry**) represents a probe result and is indexed by three variables:

- The first two variables, **pingCtlOwnerIndex** and **pingCtlTestName**, are the same ones used for **pingCtlTable**, which identifies the test.
- The third variable, **pingProbeHistoryIndex**, is a counter to uniquely identify each probe result.

The maximum number of **pingProbeHistoryTable** entries created for a given test is limited by **pingCtlMaxRows**. If **pingCtlMaxRows** is set to 0, no **pingProbeHistoryTable** entries are created for that test.

Each time a probe result is determined, a **pingProbeHistoryEntry** is created and added to **pingProbeHistoryTable**. **pingProbeHistoryIndex** of the new **pingProbeHistoryEntry** is 1 greater than the last **pingProbeHistoryEntry** added to **pingProbeHistoryTable** for that test. **pingProbeHistoryIndex** is set to 1 if this is the first entry in the table. The same test can be run multiple times, so this index keeps growing.

If **pingProbeHistoryIndex** of the last **pingProbeHistoryEntry** added is 0xFFFFFFFF, the next **pingProbeHistoryEntry** added has **pingProbeHistoryIndex** set to 1.

The following are recorded for each probe result:

- **pingProbeHistoryResponse**—Time to live (TTL)
- **pingProbeHistoryStatus**—What happened and why
- **pingProbeHistoryLastRC**—Return code (RC) value of ICMP packet
- **pingProbeHistoryTime**—Timestamp when probe result was determined

When a probe cannot be sent, **pingProbeHistoryResponse** is set to 0. When a probe times out, **pingProbeHistoryResponse** is set to the difference between the time when the probe was discovered to be timed out and the time when the probe was sent.

Generating Traps

For any trap to be generated, the appropriate bit of **pingCtlTrapGeneration** must be set. You must also configure a trap group to receive remote operations. A trap is generated under the following conditions:

- A **pingProbeFailed** trap is generated every time **pingCtlTrapProbeFailureFilter** number of consecutive probes fail during the test.
- A **pingTestFailed** trap is generated when the test completes and at least **pingCtlTrapTestFailureFilter** number of probes fail.
- A **pingTestCompleted** trap is generated when the test completes and fewer than **pingCtlTrapTestFailureFilter** probes fail.

NOTE: A probe is considered a failure when **pingProbeHistoryStatus** of the probe result is anything besides **responseReceived**.

For information about how to configure a trap group to receive remote operations, see "Configuring SNMP Trap Groups" on page 132 and "Example: Setting Trap Notification for Remote Operations" on page 187.

Gathering Ping Test Results

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

You can either poll **pingResultsOperStatus** to find out when the test is complete or request that a trap be sent when the test is complete. For more information about **pingResultsOperStatus**, see "pingResultsTable" on page 190. For more information about Ping MIB traps, see "Generating Traps" on page 192.

The statistics calculated and then stored in **pingResultsTable** include:

- **pingResultsMinRtt**—Minimum round-trip time
- **pingResultsMaxRtt**—Maximum round-trip time
- **pingResultsAverageRtt**—Average round-trip time
- **pingResultsProbeResponses**—Number of responses received
- **pingResultsSentProbes**—Number of attempts to send probes
- **pingResultsRttSumOfSquares**—Sum of squares of round-trip times
- **pingResultsLastGoodProbe**—Timestamp of the last response

You can also consult **pingProbeHistoryTable** for more detailed information about each probe. The index used for **pingProbeHistoryTable** starts at 1, goes to 0xFFFFFFFF, and wraps to 1 again.

For example, if **pingCtlProbeCount** is 15 and **pingCtlMaxRows** is 5, then upon completion of the first run of this test, **pingProbeHistoryTable** contains probes like those in Table 18 on page 193.

Table 18: Results in pingProbeHistoryTable: After the First Ping Test

pingProbeHistoryIndex	Probe Result
11	Result of 11th probe from run 1
12	Result of 12th probe from run 1
13	Result of 13th probe from run 1
14	Result of 14th probe from run 1
15	Result of 15th probe from run 1

Upon completion of the first probe of the second run of this test, **pingProbeHistoryTable** will contain probes like those in Table 19 on page 193.

Table 19: Results in pingProbeHistoryTable: After the First Probe of the Second Test

pingProbeHistoryIndex	Probe Result
12	Result of 12th probe from run 1
13	Result of 13th probe from run 1

Table 19: Results in pingProbeHistoryTable: After the First Probe of the Second Test *(continued)*

pingProbeHistoryIndex	Probe Result
14	Result of 14th probe from run 1
15	Result of 15th probe from run 1
16	Result of 1st probe from run 2

Upon completion of the second run of this test, **pingProbeHistoryTable** will contain probes like those in Table 20 on page 194.

Table 20: Results in pingProbeHistoryTable: After the Second Ping Test

pingProbeHistoryIndex	Probe Result
26	Result of 11th probe from run 2
27	Result of 12th probe from run 2
28	Result of 13th probe from run 2
29	Result of 14th probe from run 2
30	Result of 15th probe from run 2

History entries can be deleted from the MIB in two ways:

- More history entries for a given test are added and the number of history entries exceeds **pingCtlMaxRows**. The oldest history entries are deleted to make room for the new ones.
- You delete the entire test by setting **pingCtlRowStatus** to **destroy**.

Stopping a Ping Test

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

To stop an active test, set **pingCtlAdminStatus** to **disabled**. To stop the test and remove its **pingCtlEntry**, **pingResultsEntry**, and any **pingHistoryEntry** objects from the MIB, set **pingCtlRowStatus** to **destroy**.

Interpreting Ping Variables

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

This section clarifies the ranges for the following variables that are not explicitly specified in the Ping MIB:

- **pingCtlDataSize**—The value of this variable represents the total size of the payload (in bytes) of an outgoing probe packet. This payload includes the timestamp (8 bytes) that is used to time the probe. This is consistent with the definition of **pingCtlDataSize** (maximum value of 65,507) and the standard ping application.

  If the value of **pingCtlDataSize** is between 0 and 8 inclusive, it is ignored and the payload is 8 bytes (the timestamp). The Ping MIB assumes all probes are timed, so the payload must always include the timestamp.

  For example, if you wish to add an additional 4 bytes of payload to the packet, you must set **pingCtlDataSize** to 12.

- **pingCtlDataFill**—The first 8 bytes of the data segment of the packet is for the timestamp. After that, the **pingCtlDataFill** pattern is used in repetition. The default pattern (when **pingCtlDataFill** is not specified) is (00, 01, 02, 03 ... FF, 00, 01, 02, 03 ... FF, ...).
- **pingCtlMaxRows**—The maximum value is 255.
- **pingMaxConcurrentRequests**—The maximum value is 500.
- **pingCtlTrapProbeFailureFilter** and **pingCtlTrapTestFailureFilter**—A value of 0 for **pingCtlTrapProbeFailureFilter** or **pingCtlTrapTestFailureFilter** is not well defined by the Ping MIB. If **pingCtlTrapProbeFailureFilter** is 0, **pingProbeFailed** traps will not be generated for the test under any circumstances. If **pingCtlTrapTestFailureFilter** is 0, **pingTestFailed** traps will not be generated for the test under any circumstances.

Using the Traceroute MIB for Remote Monitoring Devices Running Junos OS

Supported Platforms ACX Series, M Series, MX Series, PTX Series, QFX Series, T Series

A traceroute test approximates the path packets take from the local host to the remote host.

RFC 2925 is the authoritative description of the Traceroute MIB in detail and provides the ASN.1 MIB definition of the Traceroute MIB.

Related Documentation

- SNMP Remote Operations Overview on page 185
- *Starting a Traceroute Test*
- *Monitoring a Running Traceroute Test*
- *Monitoring Traceroute Test Completion*
- *Gathering Traceroute Test Results*
- *Stopping a Traceroute Test*
- *Interpreting Traceroute Variables*

CHAPTER 9

Tracing SNMP Activity

Tracing SNMP Activity on a Device Running Junos OS

Supported Platforms ACX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

SNMP tracing operations track activity for SNMP agents and record the information in log files. The logged error descriptions provide detailed information to help you solve problems faster.

By default, Junos OS does not trace any SNMP activity. If you include the **traceoptions** statement at the **[edit snmp]** hierarchy level, the default tracing behavior is:

- Important activities are logged in files located in the **/var/log** directory. Each log is named after the SNMP agent that generates it. Currently, the following log files are created in the **/var/log** directory when the **traceoptions** statement is used:
  - chassisd
  - craftd
  - ilmid
  - mib2d
  - rmopd
  - serviced
  - snmpd
- When a trace file named ***filename*** reaches its maximum size, it is renamed ***filename*.0**, then ***filename*.1**, and so on, until the maximum number of trace files is reached. Then the oldest trace file is overwritten. (For more information about how log files are created, see the *System Log Monitoring and Troubleshooting Guide for Security Devices*.)
- Log files can be accessed only by the user who configured the tracing operation.

You cannot change the directory (**/var/log**) in which trace files are located. However, you can customize the other trace file settings by including the following statements at the **[edit snmp]** hierarchy level:

```
[edit snmp]
traceoptions {
    file <files number> <match regular-expression> <size size> <world-readable |
        no-world-readable>;
    flag flag;
    memory-trace;
    no-remote-trace;
    no-default-memory-trace;
}
```

These statements are described in the following sections:

- Configuring the Number and Size of SNMP Log Files on page 198
- Configuring Access to the Log File on page 198
- Configuring a Regular Expression for Lines to Be Logged on page 199
- Configuring the Trace Operations on page 199

Configuring the Number and Size of SNMP Log Files

By default, when the trace file reaches 128 kilobytes (KB) in size, it is renamed ***filename*.0**, then ***filename*.1**, and so on, until there are three trace files. Then the oldest trace file (***filename*.2**) is overwritten.

You can configure the limits on the number and size of trace files by including the following statements at the **[edit snmp traceoptions]** hierarchy level:

```
[edit snmp traceoptions]
file files number size size;
```

For example, set the maximum file size to 2 MB, and the maximum number of files to 20. When the file that receives the output of the tracing operation (***filename***) reaches 2 MB, ***filename*** is renamed ***filename*.0**, and a new file called ***filename*** is created. When the new ***filename*** reaches 2 MB, ***filename*.0** is renamed ***filename*.1** and ***filename*** is renamed ***filename*.0**. This process repeats until there are 20 trace files. Then the oldest file (***filename*.19**) is overwritten by the newest file (***filename*.0**).

The number of files can be from 2 through 1000 files. The file size of each file can be from 10 KB through 1 gigabyte (GB).

Configuring Access to the Log File

By default, log files can be accessed only by the user who configured the tracing operation.

To specify that any user can read all log files, include the **file world-readable** statement at the **[edit snmp traceoptions]** hierarchy level:

```
[edit snmp traceoptions]
file world-readable;
```

To explicitly set the default behavior, include the **file no-world-readable** statement at the **[edit snmp traceoptions]** hierarchy level:

```
[edit snmp traceoptions]
file no-world-readable;
```

Configuring a Regular Expression for Lines to Be Logged

By default, the trace operation output includes all lines relevant to the logged activities.

You can refine the output by including the **match** statement at the **[edit snmp traceoptions file *filename*]** hierarchy level and specifying a regular expression (regex) to be matched:

```
[edit snmp traceoptions]
file filename match regular-expression;
```

Configuring the Trace Operations

By default, only important activities are logged. You can specify which trace operations are to be logged by including the following **flag** statement (with one or more tracing flags) at the **[edit snmp traceoptions]** hierarchy level:

```
[edit snmp traceoptions]
flag {
   all;
   configuration;
   database;
   events;
   general;
   interface-stats;
   nonvolatile-sets;
   pdu;
   policy;
   protocol-timeouts;
   routing-socket;
   server;
   subagent;
   timer;
   varbind-error;
}
```

Table 21 on page 199 describes the meaning of the SNMP tracing flags.

Table 21: SNMP Tracing Flags

Flag	Description	Default Setting
all	Log all operations.	Off
configuration	Log reading of the configuration at the **[edit snmp]** hierarchy level.	Off
database	Log events involving storage and retrieval in the events database.	Off
events	Log important events.	Off

Table 21: SNMP Tracing Flags *(continued)*

Flag	Description	Default Setting
general	Log general events.	Off
interface-stats	Log physical and logical interface statistics.	Off
nonvolatile-set	Log nonvolatile SNMP set request handling.	Off
pdu	Log SNMP request and response packets.	Off
policy	Log policy processing.	Off
protocol-timeouts	Log SNMP response timeouts.	Off
routing-socket	Log routing socket calls.	Off
server	Log communication with processes that are generating events.	Off
subagent	Log subagent restarts.	Off
timer	Log internal timer events.	Off
varbind-error	Log variable binding errors.	Off

To display the end of the log for an agent, issue the **show log *agent*d | last** operational mode command:

```
[edit]
user@host# run show log agentd | last
```

where ***agent*** is the name of an SNMP agent.

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248
- Example: Tracing SNMP Activity on page 200
- *Configuring SNMP*

Example: Tracing SNMP Activity

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Trace information about SNMP packets:

```
[edit]
snmp {
   traceoptions {
      file size 10k files 5;
      flag pdu;
```

```
        flag protocol-timeouts;
        flag varbind-error;
    }
}
```

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115
- Tracing SNMP Activity on a Device Running Junos OS on page 197
- Configuration Statements at the [edit snmp] Hierarchy Level on page 248

CHAPTER 10

Configuring Vital MIB Data

Understanding Vital MIB OID Data Collection

Supported Platforms SRX Series

MIB object identifier (OID) data is collected and configured for later use in reports. You can configure data collection duration (default is 3 days), dump file size limitation (default is 5 megabytes for branch SRX Series and 10 megabytes for high-end SRX Series), and disk storage limitation (default is 80 percent). The expired dump file is removed automatically. When the dump file exceeds the limited size, a new dump file is created and the old dump file is compressed. When disk utilization exceeds the storage limitation, data collection is skipped but is attempted the next time. If an issue should arise, then the collected data is examined to help identify its cause.

Once you enable a predefined group, the vital data of all OIDs in the group are periodically collected and analyzed. Only critical data is collected when CPU utilization exceeds 60 percent but is within 80 percent.

A maximum of 64 groups per OIDs are supported for branch SRX Series devices and a maximum of 128 groups per OIDs are supported for high-end SRX Series devices.

You can also collect raw MIB OID data. For the format of raw OID output, the first volume is 40 characters and the second volume is 30 characters in length. Any extra characters are stripped.

TIP: To make the dump file easily understood, we recommend that you configure short comments for each raw OID.

Use the **set system processes system-log-vital disable** command to manually disable the syslvd process (daemon). Disabling syslvd will not impact the existing data in the dump file. Once all configuration commands are removed, syslvd is disabled automatically. If syslvd is disabled in the middle of a collection, data from the current collection will be lost but data available in the current dump file is retained.

Related Documentation

- Generating Raw MIB OID from a Policy on page 205
- Generating Readable Raw OID Data Collections on page 204

Generating Readable Raw OID Data Collections

Supported Platforms SRX Series

You can use the **set system log-vital add *oid* comment *"comment"*** command to make raw object identifiers (OIDs) that are lengthy and unreadable easily understood.

```
[edit system]
log-vital {
   add oid {
      comment comment;
   }
}
```

The **OID** parameter can be formatted as mib-table.index. For example, jnxOperating1MinLoadAvg.9.1.0.0 is an OID.

The *"comment"* parameter describes the OID. If *"comment"* is present, the comment instead of the OID is generated as the subject of the vital data.

For example, without the *"comment"* parameter, the output of the **set system log-vital add *jnxJsPolicyNumber.0*** command in the dump file is:

```
===========
jnxJsPolicyNumber.0             1
===========
```

With the *"comment"* parameter, the output of the **set system log-vital add jnxJsPolicyNumber.0 comment *"Total Policy Number"*** command in the dump file is:

```
===========
Total Policy Number              1
===========
```

NOTE: For OIDs that are temporarily unavailable, the string NA is generated for them and the system continues to get their values for every collection. In this case, the output displayed in the dump file is:

```
 ===========
Total Policy Number               NA
===========
```

Related Documentation

- Generating Raw MIB OID from a Policy on page 205

Generating Raw MIB OID from a Policy

Supported Platforms SRX Series

You can generate a raw MIB OID from a policy. You can also monitor the session number associated with the policy and other policy MIB tables.

For example, consider a policy called **test**. Monitor the session number associated with the policy.

```
[edit]
  from-zone untrust to-zone trust {
  policy test {
    match {
      source-address any;
      destination-address any;
      application any;
    }
    then {
      permit;
      count;
    }
  }
}
```

To monitor a session number associated with a policy:

1. Identify the OID of the policy's session number.

   ```
   user@host> show snmp mib walk jnxJsPolicyName | match test
   jnxJsPolicyName.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
    = test
   ```

   In the above output, the index of the policy is 7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116; the policy name is test; and the MIB table name is jnxJsPolicyName.

2. With the index, verify that both the from-zone and the to-zone match the configuration. Enter the **show snmp mib get** command.

   ```
   user@host> show snmp mib get
   jnxJsPolicyFromZone.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
   jnxJsPolicyFromZone.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
    = untrust
   ```

```
user@host> show snmp mib get
jnxJsPolicyToZone.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
jnxJsPolicyToZone.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
 = trust
```

3. Perform a mandatory from-zone and to-zone match check to avoid a scenario where there is a policy with the same name but the from-zone or the to-zone is different.
4. After performing both the from-zone and the to-zone match checks, ensure that 7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116 is the index of the policy called test in various policy MIB tables.
5. Monitor the session number using the following command:

   ```
   [edit]
   user@host# set system log-vital add
   jnxJsPolicyStatsNumSessions.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
   comment "sess num of policy test"
   ```

   The output of the configuration is:

   ```
   ===========
   sess num of policy test    100
   ===========
   ```

To monitor other policy MIB tables:

1. Combine a MIB table's name with the index.
2. Monitor the session setup rate for the test policy using the command:

   ```
   [edit]
   set system log-vital add
   jnxJsPolicyStatsSessionRate.7.117.110.116.114.117.115.116.5.116.114.117.115.116.4.116.101.115.116
   comment "sess setup rate of policy test"
   ```

   The output of the configuration is:

   ```
   ===========
   sess setup rate of policy test    233
   ===========
   ```

Related Documentation

- Understanding Vital MIB OID Data Collection on page 203
- Generating Readable Raw OID Data Collections on page 204

Generating Vital Data from a Predefined Group

Supported Platforms SRX Series

You can use the **set system log-vital group [cluster-counter | idp | operating | storage | spu *<spu-name>* | screen *<zone-name>*** command to enable a predefined group.

```
[edit system]
group {
  operating;
  idp;
  storage;
```

```
    cluster-counter;
    screen;
    spu;
}
```

NOTE: The parameter for **spu-name** must be *fwdd*, *all*, *fpcy.picz* or *nodex.fpcy.picz*.

The predefined groups are operating, SPU, storage, IDP, screen, and cluster-counter. Once a group is enabled, all OIDs in the group are periodically collected and dumped.

The operating group includes state, temperature, current CPU utilization percentage, buffer utilization percentage, heap-utilization percentage, up time, average-load in the last 1 minute, 5 minutes, or 15 minutes, and buffer-pool utilization percentage in the control plane of each operating component in the system.

The IDP group includes IDP data plane memory usage, IDP session usage and policies loaded number.

The storage group includes storage utilization of directory **/var/log**.

The cluster-counter group includes current total session number, total CPS, IPv4 CPS, IPv6 CPS, current total IPv4 session number, and current total IPv6 session number of both node 0 and node 1.

The screen group includes screen statistics of a specified zone.

The SPU group includes CPU usage, memory usage, current flow session number, current CP session number, IPv4 session number, IPv6 session number, CP IPv4 session number, and CP IPv6 session number of the SPU.

Related Documentation

- Generating Raw MIB OID from a Policy on page 205

Generating Vital Data from an Interface

Supported Platforms SRX Series

You can monitor the statistics of interface ge-0/0/0 by first obtaining the SNMP ifIndex from the interface.

```
user@host> show interfaces ge-0/0/0

Physical interface: ge-0/0/0, Enabled, Physical link is Up
  Interface index: 134, SNMP ifIndex: 509
```

In this output, the 509 value is the index of ge-0/0/0 in the interface MIB table. By combining this index value with the interface MIB tables, the vital data of the interface can be periodically collected.

For example, combine the 509 index with the ifInErrors interface MIB table to collect the In-Error data of interface ge-0/0/0 by using the following command:

```
[edit]
user@host# set system log-vital add ifInErrors.509 comment “In-Err of ge-0/0/0”
```

The output for the command is:

```
===========
In-Err of ge-0/0/0    100
===========
```

The following interface MIB tables can be used to collect vital data:

- ifInOctets
- ifInUcastPkts
- ifInNUcastPkts
- ifInDiscards
- ifInErrors
- ifInUnknownProtos
- ifOutOctets
- ifOutUcastPkts
- ifOutNUcastPkts
- ifOutDiscards
- ifOutErrors

Related Documentation

- Generating Raw MIB OID from a Policy on page 205
- Generating Readable Raw OID Data Collections on page 204

Generating Vital Data from an IPsec VPN

Supported Platforms SRX Series

You can monitor the vital data of an IPsec VPN by first obtaining the index of the VPN in the IPsec VPN MIB table.

For example, consider the following below policy-based VPN configuration, where the name of the policy is test.

```
user@host> show configuration security policies

  from-zone untrust to-zone trust {
    policy test {
      match {
        source-address any;
        destination-address any;
        application any;
      }
      then {
        permit {
          tunnel {
```

```
            ipsec-vpn ike-vpn;
          }
        }
      }
    }
  }
```

To monitor the error statistics for the VPN, you must first obtain the index of the VPN in the IPsec VPN MIB table. You can obtain this value by using the command:

```
user@host> show snmp mib walk jnxJsIpSecTunPolicyName | match test
jnxJsIpSecTunPolicyName.1.4.2.2.2.1.2 = test
```

In the output, 1.4.2.2.2.1.2 is the index of the IPsec SA associated with the policy called test. By combining the index with various IPsec VPN MIB tables, you can monitor the statistics by using the following commands:

```
[edit]
user@host# set system log-vital add jnxIpSecTunMonReplayDropPkts.1.4.2.2.2.1.2 comment
"Anti-Replay drop number of VPN policy test"
user@host# set system log-vital add jnxIpSecTunMonBadHeaders.1.4.2.2.2.1.2 comment "Bad
Header number of VPN policy test"
```

Related Documentation

- Generating Vital Data from a Screen on page 210

Generating Vital Data from a NAT Rule

Supported Platforms SRX Series

You can monitor the vital data from a NAT rule (in this example, r1) by first obtaining the MIB index of r1.

Consider the following source NAT configuration.

```
user@host> show configuration security nat

  source {
    rule-set rs1 {
      from zone trust;
      to zone untrust;
      rule r1 {
        match {
          source-address 17.0.0.0/8;
          destination-address 0.0.0.0/0;
        }
        then {
          source-nat {
            interface;
          }
        }
      }
    }
  }
```

To find the MIB index of r1, enter the following command:

```
[edit]
```

```
user@host# show snmp mib walk jnxJsNatRuleName | grep r1
jnxJsNatRuleName.2.114.49.1 = r1
```

The output shows that 2.114.49.1 is the MIB index of r1.

Therefore, by combining the index with NAT MIB table jnxJsNatRuleHits, the session number associated with NAT rule r1 can be monitored by using the command:

```
[edit]
user@host# set system log-vital add jnxJsNatRuleHits.2.114.49.1 comment "Number of sessions on NAT rule r1"
```

Related Documentation

- Generating Readable Raw OID Data Collections on page 204

Generating Vital Data from an Operating Component

Supported Platforms SRX Series

You can monitor the vital data of an operating component. For example, to monitor the temperature of the SPC component located at slot 3 of node 0, enter the following command:

```
user@host> show snmp mib walk jnxOperatingDescr | match "SPC @ 3"

jnxOperatingDescr.7.4.0.0 = node0 FPC: SRX5k SPC @ 3/*/*
jnxOperatingDescr.7.10.0.0 = node1 FPC: SRX5k SPC @ 3/*/*
```

In the output, the SPC index at slot 3 of node 0 in the operating MIB table is 7.4.0.0. By combining the 7.4.0.0 index with operating MIB table jnxOperatingTemp, the temperature of SPC at slot 3 of node 0 can be monitored by using the following command:

```
[edit]
user@host# set system log-vital add jnxOperatingTemp.7.4.0.0 comment "Temperature of node0 SPC-3"
```

Related Documentation

- Generating Vital Data from a Screen on page 210

Generating Vital Data from a Screen

Supported Platforms SRX Series

The screen group collects all screen statistics of a specified zone. However, it can only collect some of the statistics rather than all statistics.

For example, consider the following screen configuration, where the number of UDP flood attacks in the untrust zone is to be monitored.

user@host> **show configuration security screen**

```
ids-option zone-syn-flood {
    tcp {
        syn-flood {
            timeout 20;
        }
    }
```

```
}
```

user@host> **show configuration security zones**

```
security-zone untrust {
  screen zone-syn-flood;
  ...
  ...
}
```

To monitor the number of UDP flood attacks, you must first obtain the index of the untrust zone in various screen MIB tables.

```
user@host> show snmp mib walk jnxJsScreenZoneName | match untrust
jnxJsScreenZoneName.117.110.116.114.117.115.116.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
 = untrust
```

In the output, the string 117.110.116.114.117.115.116.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0 is the index of the untrust zone in the MIB table.

By combining the index with screen MIB table jnxJsScreenMonUdpFlood, the number can be monitored using the following command:

```
[edit]
user@host# set system log-vital add
jnxJsScreenMonUdpFlood.117.110.116.114.117.115.116.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
comment "Number of UDP flood attack"
```

Related Documentation

- Generating Vital Data from a NAT Rule on page 209

CHAPTER 11

SNMP FAQs

Managing Traps and Informs

Supported Platforms M Series, MX Series, SRX Series, T Series

The following sections contain a few tips on managing SNMP notifications:

Generating Traps Based on SysLog Events

Event policies can include an action that raises traps for events based on system log messages. This feature enables notification of an SNMP trap-based application when an important system log message occurs. You can convert any system log message,for which there is no corresponding trap, into a trap. If you are using network management system traps rather than system log messages to monitor your network, you can use this feature to ensure that you are notified of all the major events.

To configure a policy that raises a trap on receipt of an event, include the following statements at the **[edit event-options policy *policy-name*]** hierarchy level:

```
[edit event-options policy policy-name]
events [ events ];
then {
  raise-trap;
}
```

The following example shows the sample configuration for raising a trap for the event **ui_mgd_terminate**:

Generating Traps Based on SysLog Events

```
[edit event-options policy p1]
events ui_mgd_terminate;
then {
  raise-trap;
}
```

Filtering Traps Based on the Trap Category

SNMP traps are categorized into many categories. The Junos OS provides a configuration option, **categories** at the **[edit snmp trap-group *trap-group*]** hierarchy level, that enables you to specify categories of traps that you want to receive on a particular host. You can use this option when you want to monitor only specific modules of the Junos OS.

The following example shows a sample configuration for receiving only **link**, **vrrp-events**, **services**, and **otn-alarms** traps:

```
[edit snmp]
trap-group jnpr {
    categories {
        link;
        vrrp-events;
        services;
        otn-alarms;
    }
    targets {
        192.168.69.179;
    }
}
```

Filtering Traps Based on the Object Identifier

The Junos OS also provides a more advanced filter option that enables you to filter out specific traps based on their object identifiers. You can use the **notify-filter** option to filter out a specific trap or a group of traps.

The following example shows the sample configuration for excluding Juniper Networks enterprise-specific configuration management traps (note that the SNMPv3 configuration also supports filtering of SNMPv1 and SNMPv2 traps as is shown in the following example):

```
[edit snmp]
v3 {
    vacm {
        security-to-group {
            security-model v2c {
                security-name sn_v2c_trap {
                    group gr_v2c_trap;
                }
            }
        }
        access {
            group gr_v2c_trap {
                default-context-prefix {
                    security-model v2c {
                        security-level none {
                            read-view all;
                            notify-view all;
                        }
                    }
                }
            }
```

```
    }
  }
  target-address TA_v2c_trap {
    address 10.209.196.166;
    port 9001;
    tag-list tg1;
    target-parameters TP_v2c_trap;
  }
  target-parameters TP_v2c_trap {
    parameters {
      message-processing-model v2c;
      security-model v2c;
      security-level none;
      security-name sn_v2c_trap;
    }
    notify-filter nf1;
  }
  notify v2c_notify {
    type trap;
    tag tg1;
  }
  notify-filter nf1 {
    oid .1.3.6.1.4.1.2636.4.5 exclude;
    oid .1 include;
  }
  snmp-community index1 {
    community-name "$9$tDLl01h7Nbw2axN"; ## SECRET-DATA
    security-name sn_v2c_trap;
    tag tg1;
  }
  view all {
    oid .1 include;
  }
}
```

Related Documentation

- *Understanding SNMP Implementation in the Junos OS*
- *Configuring SNMP on Devices Running the Junos OS*
- *Monitoring SNMP Activity and Tracking Problems That Affect SNMP Performance on a Device Running the Junos OS*
- *Optimizing the Network Management System Configuration for the Best Results*
- *Configuring Options on Managed Devices for Better SNMP Response Time*
- *Using the Enterprise-Specific Utility MIB to Enhance SNMP Coverage*

PART 3

Remote Monitoring (RMON) with SNMP

CHAPTER 12

RMON Overview

Understanding RMON Alarms

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, T Series

An RMON alarm identifies:

- A specific MIB object that is monitored.
- The frequency of sampling.
- The method of sampling.
- The thresholds against which the monitored values are compared.

An RMON alarm can also identify a specific **eventTable** entry to be triggered when a threshold is crossed.

Configuration and operational values are defined in **alarmTable** in RFC 2819. Additional operational values are defined in Juniper Networks enterprise-specific extensions to **alarmTable** (**jnxRmonAlarmTable**).

This topic covers the following sections:

alarmTable

alarmTable in the RMON MIB allows you to monitor and poll the following:

- **alarmIndex**—The index value for **alarmTable** that identifies a specific entry.
- **alarmInterval**—The interval, in seconds, over which data is sampled and compared with the rising and falling thresholds.
- **alarmVariable**—The MIB variable that is monitored by the alarm entry.
- **alarmSampleType**—The method of sampling the selected variable and calculating the value to be compared against the thresholds.

- **alarmValue**—The value of the variable during the last sampling period. This value is compared with the rising and falling thresholds.
- **alarmStartupAlarm**—The alarm sent when the entry is first activated.
- **alarmRisingThreshold**—The upper threshold for the sampled variable.
- **alarmFallingThreshold**—The lower threshold for the sampled variable.
- **alarmRisingEventIndex**—The **eventTable** entry used when a rising threshold is crossed.
- **alarmFallingEventIndex**—The **eventTable** entry used when a falling threshold is crossed.
- **alarmStatus**—Method for adding and removing entries from the table. It can also be used to change the state of an entry to allow modifications.

NOTE: **If this object is not set to valid, the associated event alarm does not take any action.**

jnxRmonAlarmTable

The **jnxRmonAlarmTable** is a Juniper Networks enterprise-specific extension to **alarmTable**. It provides additional operational information and includes the following objects:

- **jnxRmonAlarmGetFailCnt**—The number of times the internal **Get** request for the variable monitored by this entry has failed.
- **jnxRmonAlarmGetFailTime**—The value of **sysUpTime** when an internal **Get** request for the variable monitored by this entry last failed.
- **jnxRmonAlarmGetFailReason**—The reason an internal **Get** request for the variable monitored by this entry last failed.
- **jnxRmonAlarmGetOkTime**—The value of **sysUpTime** when an internal **Get** request for the variable monitored by this entry succeeded and the entry left the **getFailure** state.
- **jnxRmonAlarmState**—The current state of this RMON alarm entry.

To view the Juniper Networks enterprise-specific extensions to the RMON Events and Alarms and Event MIB, see http://www.juniper.net/techpubs/en_US/junos10.3/topics/reference/mibs/mib-jnx-rmon.txt.

For more information about the Juniper Networks enterprise-specific extensions to the RMON Events and Alarms MIB, see "*RMON Events and Alarms MIB*" in the *SNMP MIBs and Traps Monitoring and Troubleshooting Guide for Security Devices*.

Related Documentation

- Understanding RMON Events on page 221
- Configuring an Alarm Entry and Its Attributes on page 224
- *Using alarmTable to Monitor MIB Objects*

Understanding RMON Events

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, T Series

An RMON event allows you to log the crossing of thresholds of other MIB objects. It is defined in **eventTable** for the RMON MIB.

This section covers the following topics:

- eventTable on page 221

eventTable

eventTable contains the following objects:

- **eventIndex**—An index that uniquely identifies an entry in **eventTable**. Each entry defines one event that is generated when the appropriate conditions occur.
- **eventDescription**—A comment describing the event entry.
- **eventType**—Type of notification that the probe makes about this event.
- **eventCommunity**—Trap group used if an SNMP trap is to be sent. If **eventCommunity** is not configured, a trap is sent to each trap group configured with the **rmon-alarm** category.
- **eventLastTimeSent**—Value of **sysUpTime** when this event entry last generated an event.
- **eventOwner**—Any text string specified by the creating management application or the command-line interface (CLI). Typically, it is used to identify a network manager (or application) and can be used for fine access control between participating management applications.
- **eventStatus**—Status of this event entry.

NOTE: If this object is not set to valid, no action is taken by the associated event entry. When this object is set to valid, all previous log entries associated with this entry (if any) are deleted.

Related Documentation

- Understanding RMON Alarms on page 219
- Configuring an Event Entry and Its Attributes on page 228

CHAPTER 13

Configuring RMON Alarms and Events

Understanding RMON Alarms and Events Configuration

Supported Platforms ACX Series, LN Series, M Series, MX Series, T Series

Junos OS supports monitoring routers from remote devices. These values are measured against thresholds and trigger events when the thresholds are crossed. You configure remote monitoring (RMON) alarm and event entries to monitor the value of a MIB object.

To configure RMON alarm and event entries, you include statements at the **[edit snmp]** hierarchy level of the configuration:

```
[edit snmp]
rmon {
  alarm index {
    description text-description;
    falling-event-index index;
    falling-threshold integer;
    falling-threshold-interval seconds;
    interval seconds;
    rising-event-index index;
    rising-threshold integer;
    request-type (get-next-request | get-request | walk-request);
    sample-type (absolute-value | delta-value);
    startup-alarm (falling-alarm | rising-alarm | rising-or-falling-alarm);
    syslog-subtag syslog-subtag;
    variable oid-variable;
    event index {
      community community-name;
      description description;
      type type;
    }
  }
}
```

Related Documentation

- Understanding RMON Alarms on page 219
- Understanding RMON Events on page 221
- Configuring an Alarm Entry and Its Attributes on page 224
- Configuring an Event Entry and Its Attributes on page 228

Configuring an Alarm Entry and Its Attributes

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

An alarm entry monitors the value of a MIB variable. You can configure how often the value is sampled, the type of sampling to perform, and what event to trigger if a threshold is crossed.

This section discusses the following topics:

- Configuring the Alarm Entry on page 224
- Configuring the Description on page 225
- Configuring the Falling Event Index or Rising Event Index on page 225
- Configuring the Falling Threshold or Rising Threshold on page 225
- Configuring the Interval on page 226
- Configuring the Falling Threshold Interval on page 226
- Configuring the Request Type on page 226
- Configuring the Sample Type on page 227
- Configuring the Startup Alarm on page 227
- Configuring the System Log Tag on page 227
- Configuring the Variable on page 228

Configuring the Alarm Entry

An alarm entry monitors the value of a MIB variable. The **rising-event-index**, **rising-threshold**, **sample-type**, and **variable** statements are mandatory. All other statements are optional.

To configure the alarm entry, include the **alarm** statement and specify an index at the **[edit snmp rmon]** hierarchy level:

```
[edit snmp rmon]
alarm index {
  description description;
  falling-event-index index;
  falling-threshold integer;
  falling-threshold-interval seconds;
  interval seconds;
  rising-event-index index;
  rising-threshold integer;
  sample-type (absolute-value | delta-value);
  startup-alarm (falling-alarm | rising alarm | rising-or-falling-alarm);
```

```
    variable oid-variable;
}
```

index is an integer that identifies an alarm or event entry.

Configuring the Description

The description is a text string that identifies the alarm entry.

To configure the description, include the **description** statement and a description of the alarm entry at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
description description;
```

Configuring the Falling Event Index or Rising Event Index

The falling event index identifies the event entry that is triggered when a falling threshold is crossed. The rising event index identifies the event entry that is triggered when a rising threshold is crossed.

To configure the falling event index or rising event index, include the **falling-event-index** or **rising-event-index** statement and specify an index at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
falling-event-index index;
rising-event-index index;
```

index can be from 0 through 65,535. The default for both the falling and rising event index is 0.

Configuring the Falling Threshold or Rising Threshold

The falling threshold is the lower threshold for the monitored variable. When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is 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 startup alarm is equal to **falling-alarm** or **rising-or-falling-alarm**. After a falling event is generated, another falling event cannot be generated until the sampled value rises above this threshold and reaches the rising threshold. You must specify the falling threshold as an integer. Its default is 20 percent less than the rising threshold.

By default, the rising threshold is 0. The rising threshold is the upper threshold for the monitored variable. When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is 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 **startup-alarm** is equal to **rising-alarm** or **rising-or-falling-alarm**. After a rising event is generated, another rising event cannot be generated until the sampled value falls below this threshold and reaches the falling threshold. You must specify the rising threshold as an integer.

To configure the falling threshold or rising threshold, include the **falling-threshold** or **rising-threshold** statement at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
falling-threshold integer;
rising-threshold integer;
```

integer can be a value from -2,147,483,647 through 2,147,483,647.

Configuring the Interval

The interval represents the period of time, in seconds, over which the monitored variable is sampled and compared with the rising and falling thresholds.

To configure the interval, include the **interval** statement and specify the number of seconds at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
interval seconds;
```

seconds can be a value from 1 through 2,147,483,647. The default is 60 seconds.

Configuring the Falling Threshold Interval

The falling threshold interval represents the interval between samples when the rising threshold is crossed. Once the alarm crosses the falling threshold, the regular sampling interval is used.

NOTE: **You cannot configure the falling threshold interval for alarms that have the request type set to walk-request.**

To configure the falling threshold interval, include the **falling-threshold interval** statement at the **[edit snmp rmon alarm *index*]** hierarchy level and specify the number of seconds:

```
[edit snmp rmon alarm index]
falling-threshold-interval seconds;
```

seconds can be a value from 1 through 2,147,483,647. The default is 60 seconds.

Configuring the Request Type

By default an RMON alarm can monitor only one object instance (as specified in the configuration). You can configure a **request-type** statement to extend the scope of the RMON alarm to include all object instances belonging to a MIB branch or to include the next object instance after the instance specified in the configuration.

To configure the request type, include the **request-type** statement at the **[edit snmp rmon alarm *index*]** hierarchy level and specify **get-next-request**, **get-request**, or **walk-request**:

```
[edit snmp rmon alarm index]
request-type (get-next-request | get-request | walk-request);
```

walk extends the RMON alarm configuration to all object instances belonging to a MIB branch. **next** extends the RMON alarm configuration to include the next object instance after the instance specified in the configuration.

Configuring the Sample Type

The sample type identifies the method of sampling the selected variable and calculating the value to be compared against the thresholds. If the value of this object is **absolute-value**, 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 **delta-value**, the value of the selected variable at the last sample is subtracted from the current value, and the difference is compared with the thresholds.

To configure the sample type, include the **sample-type** statement and specify the type of sample at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
sample-type (absolute-value | delta-value);
```

- **absolute-value**—Actual value of the selected variable is compared against the thresholds.
- **delta-value**—Difference between samples of the selected variable is compared against the thresholds.

Configuring the Startup Alarm

The startup alarm identifies the type of alarm that can be sent when this entry is first activated. You can specify it as **falling-alarm**, **rising-alarm**, or **rising-or-falling-alarm**.

To configure the startup alarm, include the **startup-alarm** statement and specify the type of alarm at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
startup-alarm (falling-alarm | rising-alarm | rising-or-falling-alarm);
```

- **falling-alarm**—Generated if the first sample after the alarm entry becomes active is less than or equal to the falling threshold.
- **rising-alarm**—Generated if the first sample after the alarm entry becomes active is greater than or equal to the rising threshold.
- **rising-or-falling-alarm**—Generated if the first sample after the alarm entry becomes active satisfies either of the corresponding thresholds.

The default is **rising-or-falling-alarm**.

Configuring the System Log Tag

The **syslog-subtag** statement specifies the tag to be added to the system log message. You can specify a string of not more than 80 uppercase characters as the system log tag.

To configure the system log tag, include the **syslog-subtag** statement at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
syslog-subtag syslog-subtag;
```

Configuring the Variable

The variable identifies the MIB object that is being monitored.

To configure the variable, include the **variable** statement and specify the object identifier or object name at the **[edit snmp rmon alarm *index*]** hierarchy level:

```
[edit snmp rmon alarm index]
variable oid-variable;
```

oid-variable is a dotted decimal (for example, **1.3.6.1.2.1.2.1.2.2.1.10.1**) or MIB object name (for example, **ifInOctets.1**).

Configuring an Event Entry and Its Attributes

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

An event entry generates a notification for an alarm entry when its rising or falling threshold is crossed. You can configure the type of notification that is generated. To configure the event entry, include the **event** statement at the **[edit snmp rmon]** hierarchy level. All statements except the **event** statement are optional.

```
[edit snmp rmon]
event index {
  community community-name;
  description description;
  type type;
}
```

index identifies an entry event.

community-name is the trap group that is used when generating a trap. If that trap group has the **rmon-alarm** trap category configured, a trap is sent to all the targets configured for that trap group. The community string in the trap matches the name of the trap group. If nothing is configured, all the trap groups are examined, and traps are sent using each group with the **rmon-alarm** category set.

description is a text string that identifies the entry.

The ***type*** variable of an event entry specifies where the event is to be logged. You can specify the type as one of the following:

- **log**—Adds the event entry to the **logTable**.
- **log-and-trap**—Sends an SNMP trap and creates a log entry.
- **none**—Sends no notification.
- **snmptrap**—Sends an SNMP trap.

The default for the event entry type is **log-and-trap**.

Related Documentation

- Understanding RMON Alarms and Events Configuration on page 223
- Understanding RMON Alarms on page 219

- Understanding RMON Events on page 221
- Configuring an Alarm Entry and Its Attributes on page 224
- Example: Configuring an RMON Alarm and Event Entry on page 229

Example: Configuring an RMON Alarm and Event Entry

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

Configure an RMON alarm and event entry:

```
[edit snmp]
rmon {
   alarm 100 {
      description “input traffic on fxp0”;
      falling-event-index 100;
      falling-threshold 10000;
      interval 60;
      rising-event-index 100;
      rising-threshold 100000;
      sample-type delta-value;
      startup-alarm rising-or-falling-alarm;
      variable ifInOctets.1;
   }
   event 100 {
      community bedrock;
      description” emergency events”;
      type log-and-trap;
   }
}
```

Related Documentation

- Understanding RMON Alarms and Events Configuration on page 223
- Configuring an Alarm Entry and Its Attributes on page 224
- Configuring an Event Entry and Its Attributes on page 228

Example: Configuring Health Monitoring

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

Configure the health monitor:

```
[edit snmp]
health-monitor {
   falling-threshold 85;
   interval 600;
   rising-threshold 75;
}
```

In this example, the sampling interval is every **600** seconds (10 minutes), the falling threshold is **85** percent of the maximum possible value for each object instance monitored, and the rising threshold is **75** percent of the maximum possible value for each object instance monitored.

Related Documentation

- Configuring Health Monitoring on Devices Running Junos OS on page 239

CHAPTER 14

Monitoring RMON Alarms and Events

Understanding RMON for Monitoring Service Quality

Supported Platforms ACX Series, LN Series, M Series, MX Series, PTX Series, T Series

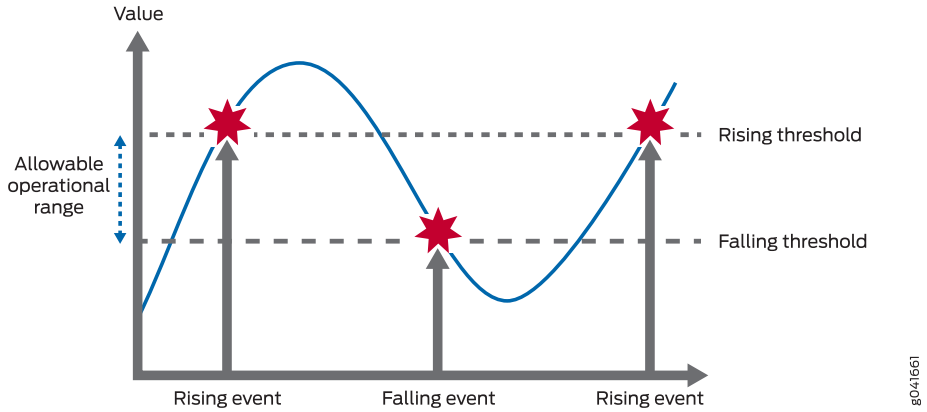
Health and performance monitoring can benefit from the remote monitoring of SNMP variables by the local SNMP agents running on each router. The SNMP agents compare MIB values against predefined thresholds and generate exception alarms without the need for polling by a central SNMP management platform. This is an effective mechanism for proactive management, as long as the thresholds have baselines determined and set correctly. For more information, see RFC 2819, *Remote Network Monitoring MIB*.

This topic includes the following sections:

Setting Thresholds

By setting a rising and a falling threshold for a monitored variable, you can be alerted whenever the value of the variable falls outside of the allowable operational range. (See Figure 3 on page 232.)

Figure 3: Setting Thresholds



Events are only generated when the threshold is first crossed in any one direction rather than after each sample period. For example, if a rising threshold crossing event is raised, no more threshold crossing events will occur until a corresponding falling event. This considerably reduces the quantity of alarms that are produced by the system, making it easier for operations staff to react when alarms do occur.

To configure remote monitoring, specify the following pieces of information:

- The variable to be monitored (by its SNMP object identifier)
- The length of time between each inspection
- A rising threshold
- A falling threshold
- A rising event
- A falling event

Before you can successfully configure remote monitoring, you should identify what variables need to be monitored and their allowable operational range. This requires some period of baselining to determine the allowable operational ranges. An initial baseline period of at least three months is not unusual when first identifying the operational ranges and defining thresholds, but baseline monitoring should continue over the life span of each monitored variable.

RMON Command-Line Interface

Junos OS provides two mechanisms you use to control the Remote Monitoring agent on the router: command-line interface (CLI) and SNMP. To configure an RMON entry using the CLI, include the following statements at the **[edit snmp]** hierarchy level:

```
rmon {
    alarm index {
        description;
        falling-event-index;
        falling-threshold;
        intervals;
        rising-event-index;
```

```
        rising-threshold;
        sample-type (absolute-value | delta-value);
        startup-alarm (falling | rising | rising-or-falling);
        variable;
      }
      event index {
        community;
        description;
        type (log | trap | log-and-trap | none);
      }
    }
```

If you do not have CLI access, you can configure remote monitoring using the SNMP Manager or management application, assuming SNMP access has been granted. (See Table 22 on page 233.) To configure RMON using SNMP, perform SNMP **Set** requests to the RMON event and alarm tables.

RMON Event Table

Set up an event for each type that you want to generate. For example, you could have two generic events, *rising* and *falling*, or many different events for each variable that is being monitored (for example, *temperature rising* event, *temperature falling* event, *firewall hit* event, *interface utilization* event, and so on). Once the events have been configured, you do not need to update them.

Table 22: RMON Event Table

Field	Description
eventDescription	Text description of this event
eventType	Type of event (for example, **log**, **trap**, or **log** and **trap**)
eventCommunity	Trap group to which to send this event (as defined in the Junos OS configuration, which is not the same as the community)
eventOwner	Entity (for example, **manager**) that created this event
eventStatus	Status of this row (for example, **valid**, **invalid**, or **createRequest**)

RMON Alarm Table

The RMON alarm table stores the SNMP object identifiers (including their instances) of the variables that are being monitored, together with any rising and falling thresholds and their corresponding event indexes. To create an RMON request, specify the fields shown in Table 23 on page 233.

Table 23: RMON Alarm Table

Field	Description
alarmStatus	Status of this row (for example, **valid**, **invalid**, or **createRequest**)

Table 23: RMON Alarm Table *(continued)*

Field	Description
alarmInterval	Sampling period (in seconds) of the monitored variable
alarmVariable	OID (and instance) of the variable to be monitored
alarmValue	Actual value of the sampled variable
alarmSampleType	Sample type (**absolute** or **delta** changes)
alarmStartupAlarm	Initial alarm (**rising**, **falling**, or **either**)
alarmRisingThreshold	Rising threshold against which to compare the value
alarmFallingThreshold	Falling threshold against which to compare the value
alarmRisingEventIndex	Index (row) of the rising event in the event table
alarmFallingEventIndex	Index (row) of the falling event in the event table

Both the **alarmStatus** and **eventStatus** fields are **entryStatus** primitives, as defined in RFC 2579, *Textual Conventions for SMIv2*.

Troubleshooting RMON

You troubleshoot the RMON agent, **rmopd**, that runs on the router by inspecting the contents of the Juniper Networks enterprise RMON MIB, **jnxRmon**, which provides the extensions listed in Table 24 on page 234 to the RFC 2819 **alarmTable**.

Table 24: jnxRmon Alarm Extensions

Field	Description
jnxRmonAlarmGetFailCnt	Number of times the internal **Get** request for the variable failed
jnxRmonAlarmGetFailTime	Value of **sysUpTime** when the last failure occurred
jnxRmonAlarmGetFailReason	Reason why the **Get** request failed
jnxRmonAlarmGetOkTime	Value of **sysUpTime** when the variable moved out of failure state
jnxRmonAlarmState	Status of this alarm entry

Monitoring the extensions in this table provides clues as to why remote alarms may not behave as expected.

Related Documentation

- Understanding Measurement Points, Key Performance Indicators, and Baseline Values on page 235

Understanding Measurement Points, Key Performance Indicators, and Baseline Values

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

This chapter topic provides guidelines for monitoring the service quality of an IP network. It describes how service providers and network administrators can use information provided by Juniper Networks routers to monitor network performance and capacity. You should have a thorough understanding of the SNMP and the associated MIB supported by Junos OS.

NOTE: For a good introduction to the process of monitoring an IP network, see RFC 2330, *Framework for IP Performance Metrics*.

This topic contains the following sections:

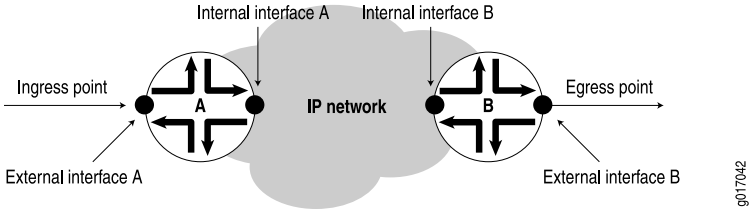
Measurement Points

Defining the measurement points where metrics are measured is equally as important as defining the metrics themselves. This section describes measurement points within the context of this chapter and helps identify where measurements can be taken from a service provider network. It is important to understand exactly where a measurement point is. Measurement points are vital to understanding the implication of what the actual measurement means.

An IP network consists of a collection of routers connected by physical links that are all running the Internet Protocol. You can view the network as a collection of routers with an ingress (entry) point and an egress (exit) point. See Figure 4 on page 235.

- Network-centric measurements are taken at measurement points that most closely map to the ingress and egress points for the network itself. For example, to measure delay across the provider network from Site A to Site B, the measurement points should be the ingress point to the provider network at Site A and the egress point at Site B.
- Router-centric measurements are taken directly from the routers themselves, but be careful to ensure that the correct router subcomponents have been identified in advance.

Figure 4: Network Entry Points



NOTE: Figure 4 on page 235 does not show the client networks at customer premises, but they would be located on either side of the ingress and egress points. Although this chapter does not discuss how to measure network services as perceived by these client networks, you can use measurements taken for the service provider network as input into such calculations.

Basic Key Performance Indicators

For example, you could monitor a service provider network for three basic key performance indicators (KPIs):

- *Availability* measures the "reachability" of one measurement point from another measurement point at the network layer (for example, using ICMP ping). The underlying routing and transport infrastructure of the provider network will support the availability measurements, with failures highlighted as unavailability.
- *Health* measures the number and type of errors that are occurring on the provider network, and can consist of both router-centric and network-centric measurements, such as hardware failures or packet loss.
- *Performance* of the provider network measures how well it can support IP services (for example, in terms of delay or utilization).

Setting Baselines

How well is the provider network performing? We recommend an initial three-month period of monitoring to identify a network's normal operational parameters. With this information, you can recognize exceptions and identify abnormal behavior. You should continue baseline monitoring for the lifetime of each measured metric. Over time, you must be able to recognize performance trends and growth patterns.

Within the context of this chapter, many of the metrics identified do not have an allowable operational range associated with them. In most cases, you cannot identify the allowable operational range until you have determined a baseline for the actual variable on a specific network.

Related Documentation

- Understanding RMON for Monitoring Service Quality on page 231
- *Defining and Measuring Network Availability*
- *Measuring Health*
- *Measuring Performance*

PART 4

Health Monitoring with SNMP

CHAPTER 15

Configuring Health Monitoring

Configuring Health Monitoring on Devices Running Junos OS

Supported Platforms LN Series, M Series, MX Series, PTX Series, T Series

As the number of devices managed by a typical network management system (NMS) grows and the complexity of the devices themselves increases, it becomes increasingly impractical for the NMS to use polling to monitor the devices. A more scalable approach is to rely on network devices to notify the NMS when something requires attention.

On Juniper Networks routers, RMON alarms and events provide much of the infrastructure needed to reduce the polling overhead from the NMS. However, with this approach, you must set up the NMS to configure specific MIB objects into RMON alarms. This often requires device-specific expertise and customizing of the monitoring application. In addition, some MIB object instances that need monitoring are set only at initialization or change at runtime and cannot be configured in advance.

To address these issues, the health monitor extends the RMON alarm infrastructure to provide predefined monitoring for a selected set of object instances (for file system usage, CPU usage, and memory usage) and includes support for unknown or dynamic object instances (such as Junos OS processes).

Health monitoring is designed to minimize user configuration requirements. To configure health monitoring entries, include the **health-monitor** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
health-monitor {
    falling-threshold percentage;
    interval seconds;
    rising-threshold percentage;
}
```

You can use the **show snmp health-monitor** operational command to view information about health monitor alarms and logs.

This topic describes the minimum required configuration and discusses the following tasks for configuring the health monitor:

Monitored Objects

When you configure the health monitor, monitoring information for certain object instances is available, as shown in Table 25 on page 240.

Table 25: Monitored Object Instances

Object	Description
jnxHrStoragePercentUsed.1	Monitors the following file system on the router or switch: **/dev/ad0s1a:** This is the root file system mounted on /.
jnxHrStoragePercentUsed.2	Monitors the following file system on the router or switch: **/dev/ad0s1e:** This is the configuration file system mounted on **/config**.
jnxOperatingCPU (RE0) **jnxOperatingCPU (RE1)**	Monitors CPU usage for Routing Engines (**RE0** and **RE1**). The index values assigned to Routing Engines depend on whether the Chassis MIB uses a zero-based or ones-based indexing scheme. Because the indexing scheme is configurable, the proper index is determined when the router or switch is initialized and when there is a configuration change. If the router or switch has only one Routing Engine, the alarm entry monitoring **RE1** is removed after five failed attempts to obtain the CPU value.
jnxOperatingBuffer (RE0) **jnxOperatingBuffer (RE1)**	Monitors the amount of memory available on Routing Engines (**RE0** and **RE1**). Because the indexing of this object is identical to that used for **jnxOperatingCPU**, index values are adjusted depending on the indexing scheme used in the Chassis MIB. As with **jnxOperatingCPU**, the alarm entry monitoring **RE1** is removed if the router or switch has only one Routing Engine.
sysApplElmtRunCPU	Monitors the CPU usage for each Junos OS process (also called daemon). Multiple instances of the same process are monitored and indexed separately.
sysApplElmtRunMemory	Monitors the memory usage for each Junos OS process. Multiple instances of the same process are monitored and indexed separately.

Minimum Health Monitoring Configuration

To enable health monitoring on the router or switch, include the **health-monitor** statement at the **[edit snmp]** hierarchy level:

```
[edit snmp]
health-monitor;
```

Configuring the Falling Threshold or Rising Threshold

The falling threshold is the lower threshold (expressed as a percentage of the maximum possible value) for the monitored variable. When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is 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. After a falling event is generated, another falling event cannot be generated until the sampled value rises above this threshold and reaches the rising threshold. You must specify the falling threshold as a percentage of the maximum possible value. The default is **70** percent.

By default, the rising threshold is **80** percent of the maximum possible value for the monitored object instance. The rising threshold is the upper threshold for the monitored variable. When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is 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. After a rising event is generated, another rising event cannot be generated until the sampled value falls below this threshold and reaches the falling threshold. You must specify the rising threshold as a percentage of the maximum possible value for the monitored variable.

To configure the falling threshold or rising threshold, include the **falling-threshold** or **rising-threshold** statement at the **[edit snmp health-monitor]** hierarchy level:

```
[edit snmp health-monitor]
falling-threshold percentage;
rising-threshold percentage;
```

percentage can be a value from **1** through **100**.

The falling and rising thresholds apply to all object instances monitored by the health monitor.

Configuring the Interval

The interval represents the period of time, in seconds, over which the object instance is sampled and compared with the rising and falling thresholds.

To configure the interval, include the **interval** statement and specify the number of seconds at the **[edit snmp health-monitor]** hierarchy level:

```
[edit snmp health-monitor]
interval seconds;
```

seconds can be a value from **1** through **2147483647**. The default is **300** seconds (5 minutes).

Log Entries and Traps

The system log entries generated for any health monitor events (thresholds crossed, errors, and so on) have a corresponding **HEALTHMONITOR** tag rather than a generic **SNMPD_RMON_EVENTLOG** tag. However, the health monitor sends generic RMON **risingThreshold** and **fallingThreshold** traps.

Related Documentation

- Understanding RMON Alarms and Events Configuration on page 223
- Configuring an Alarm Entry and Its Attributes on page 224
- Configuring an Event Entry and Its Attributes on page 228
- Example: Configuring Health Monitoring on page 229
- Understanding Device Management Functions in Junos OS on page 3

PART 5

Configuration Statements and Operational Commands

CHAPTER 16

Configuration Statements

Configuration Statements at the [edit snmp] Hierarchy Level

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

This topic shows all possible configuration statements at the **[edit snmp]** hierarchy level and their level in the configuration hierarchy. When you are configuring Junos OS, your current hierarchy level is shown in the banner on the line preceding the **user@host#** prompt.

```
[edit]
snmp {
  client-list client-list-name {
    ip-addresses;
  }
  community community-name {
    authorization authorization;
    client-list-name client-list-name;
    clients {
      address <restrict>;
    }
    logical-system logical-system-name {
      routing-instance routing-instance-name;
      clients {
        address <restrict>;
      }
    }
    routing-instance routing-instance-name {
      clients {
        address <restrict>;
      }
    }
    view view-name;
  }
  contact contact;
  description description;
  engine-id {
    (local engine-id | use-default-ip-address | use-mac-address);
  }
  filter-duplicates;
  interface [ interface-names ];
  location location;
  name name;
  nonvolatile {
    commit-delay seconds;
  }
```

```
rmon {
  alarm index {
    description description;
    falling-event-index index;
    falling-threshold integer;
    falling-threshold-interval seconds;
    interval seconds;
    request-type (get-next-request | get-request | walk-request);
    rising-event-index index;
    rising-threshold integer;
    sample-type type;
    startup-alarm alarm;
    syslog-subtag syslog-subtag;
    variable oid-variable;
  }
  event index {
    community community-name;
    description description;
    type type;
  }
}
traceoptions {
  file filename <files number> <size size> <world-readable | no-world-readable> <match
    regular-expression>;
  flag flag;
}
trap-group group-name {
  categories {
    category;
  }
  destination-port port-number;
  routing-instance instance;
  logical-system logical-system-name;
  targets {
    address;
  }
  version (all | v1 | v2);
}
trap-options {
  agent-address outgoing-interface;
  source-address address;
  enterprise-oid;
  logical-system logical-system-name {
    routing-instance routing-instance-name {
      source-address address;
    }
  }
  routing-instance routing-instance-name {
    source-address address;
  }
}
v3 {
  notify name {
    tag tag-name;
    type (trap | inform);
  }
```

```
notify-filter profile-name {
  oid oid (include | exclude);
}
snmp-community community-index {
  community-name community-name;
  security-name security-name;
  tag tag-name;
}
target-address target-address-name {
  address address;
  address-mask address-mask;
  logical-system logical-system;
  port port-number;
  retry-count number;
  routing-instance instance;
  tag-list tag-list;
  target-parameters target-parameters-name;
  timeout seconds;
}
target-parameters target-parameters-name {
  notify-filter profile-name;
  parameters {
    message-processing-model (v1 | v2c | v3);
    security-level (authentication | none | privacy);
    security-model (usm | v1 | v2c);
    security-name security-name;
  }
}
usm {
  local-engine {
    user username {
      authentication-md5 {
        authentication-password authentication-password;
      }
      authentication-none;
      authentication-sha {
        authentication-password authentication-password;
      }
      privacy-3des {
        privacy-password privacy-password;
      }
      privacy-aes128 {
        privacy-password privacy-password;
      }
      privacy-des {
        privacy-password privacy-password;
      }
      privacy-none;
    }
  }
}
vacm {
  access {
    group group-name {
      (default-context-prefix | context-prefix context-prefiix){
        security-model (any | usm | v1 | v2c) {
```

```
                security-level (authentication | none | privacy) {
                  notify-view view-name;
                  read-view view-name;
                  write-view view-name;
                }
              }
            }
          }
        }
        security-to-group {
          security-model (usm | v1 | v2c) {
            security-name security-name {
              group group-name;
            }
          }
        }
      }
    }
    view view-name {
      oid object-identifier (include | exclude);
    }
}
```

Related Documentation

- Understanding the SNMP Implementation in Junos OS on page 9
- Configuring SNMP on a Device Running Junos OS on page 115

Complete SNMPv3 Configuration Statements

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

To configure SNMPv3, include the following statements at the **[edit snmp v3]** and **[edit snmp]** hierarchy levels:

```
[edit snmp]
engine-id {
  (local engine-id | use-mac-address | use-default-ip-address);
}
view view-name {
  oid object-identifier (include | exclude);
}
[edit snmp v3]
notify name {
  tag tag-name;
  type (trap | inform);
}
notify-filter profile-name {
  oid object-identifier (include | exclude);
}
snmp-community community-index {
  community-name community-name;
  security-name security-name;
  tag tag-name;
}
target-address target-address-name {
```

```
    address address;
    address-mask address-mask;
    logical-system logical-system;
    port port-number;
    retry-count number;
    routing-instance instance;
    tag-list tag-list;
    target-parameters target-parameters-name;
    timeout seconds;
  }
  target-parameters target-parameters-name {
    notify-filter profile-name;
    parameters {
      message-processing-model (v1 | v2c | v3);
      security-level (authentication | none | privacy);
      security-model (usm | v1 | v2c);
      security-name security-name;
    }
  }
  usm {
    (local-engine | remote-engine engine-id) {
      user username {
        authentication-md5 {
          authentication-password authentication-password;
        }
        authentication-none;
        authentication-sha {
          authentication-password authentication-password;
        }
        privacy-3des {
          privacy-password privacy-password;
        }
        privacy-aes128 {
          privacy-password privacy-password;
        }
        privacy-des {
          privacy-password privacy-password;
        }
        privacy-none;
      }
    }
  }
  vacm {
    access {
      group group-name {
        (default-context-prefix | context-prefix context-prefix){
          security-model (any | usm | v1 | v2c) {
            security-level (authentication | none | privacy) {
              notify-view view-name;
              read-view view-name;
              write-view view-name;
            }
          }
        }
      }
    }
```

```
    security-to-group {
      security-model (usm | v1 | v2c) {
        security-name security-name {
          group group-name;
        }
      }
    }
  }
```

Related Documentation

- Creating SNMPv3 Users on page 138
- Configuring MIB Views on page 126
- Defining Access Privileges for an SNMP Group on page 148
- Configuring SNMPv3 Traps on a Device Running Junos OS on page 158
- Configuring SNMP Informs on page 157
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

access-list

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
[edit snmp]
  routing-instance-access {
    access-list {
      routing-instance;
      routing-instance restrict;
    }
  }
```

Hierarchy Level [edit snmp routing-instance-access]

Release Information Statement introduced in Junos OS Release 8.4.

Description Create access lists to control SNMP agents in routing instances from accessing SNMP information. To enable the SNMP agent on a routing instance to access SNMP information, specify the routing instance name. To disable the SNMP agent on a routing instance from accessing SNMP information, include the routing-instance name followed by the **restrict** keyword.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- routing-instance-access on page 310

address

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax address *address*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Specify the SNMP target address.

Options ***address***—IPv4 address of the system to receive traps or informs. You must specify an address, not a hostname.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Address on page 161

address-mask

Supported Platforms ACX Series, EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax address-mask *address-mask*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 on the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Verify the source addresses for a group of target addresses.

Options ***address-mask*** combined with the address defines a range of addresses.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Address Mask on page 162

agent-address

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax agent-address outgoing-interface;

Hierarchy Level [edit snmp trap-options]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Set the agent address of all SNMPv1 traps generated by this router or switch. Currently, the only option is **outgoing-interface**, which sets the agent address of each SNMPv1 trap to the address of the outgoing interface of that trap.

Options

outgoing-interface—Value of the agent address of all SNMPv1 traps generated by this router or switch. The **outgoing-interface** option sets the agent address of each SNMPv1 trap to the address of the outgoing interface of that trap.

Default: disabled (the agent address is not specified in SNMPv1 traps).

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Agent Address for SNMP Traps on page 131

alarm (SNMP RMON)

Supported Platforms ACX Series, EX Series, M Series, MX Series, OCX1100, PTX Series, QFX Series standalone switches, SRX210, SRX3400, T Series

Syntax

```
alarm index {
  description description;
  falling-event-index index;
  falling-threshold integer;
  falling-threshold-interval seconds;
  interval seconds;
  request-type (get-next-request | get-request | walk-request);
  rising-event-index index;
  rising-threshold integer;
  sample-type (absolute-value | delta-value);
  startup-alarm (falling-alarm | rising-alarm | rising-or-falling alarm);
  syslog-subtag syslog-subtag;
  variable oid-variable;
}
```

Hierarchy Level [edit snmp rmon]

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Configure RMON alarm entries.

Options *index*—Identifies this alarm entry as an integer.

The remaining statements are explained separately.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring an Alarm Entry and Its Attributes on page 224
- event on page 272
- *RMON MIB Event, Alarm, Log, and History Control Tables*
- *Monitoring RMON MIB Tables*
- *Understanding RMON*

authentication-md5

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
authentication-md5 {
    authentication-password authentication-password;
}
```

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user username]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure MD5 as the authentication type for the SNMPv3 user.

NOTE: You can only configure one authentication type for each SNMPv3 user.

The remaining statement is explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring MD5 Authentication on page 145

authentication-none

Supported Platforms ACX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax authentication-none;

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user *username*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure that there should be no authentication for the SNMPv3 user.

NOTE: You can configure only one authentication type for each SNMPv3 user.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring No Authentication on page 146

authentication-password

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax authentication-password *authentication-password*;

Hierarchy Level [edit snmp v3 usm local-engine user *username* authentication-md5],
[edit snmp v3 usm local-engine user *username* authentication-sha],
[edit snmp v3 usm remote-engine *engine-id* user *username* authentication-md5],
[edit snmp v3 usm remote-engine *engine-id* user *username* authentication-sha]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the password for user authentication.

Options ***authentication-password***—Password that a user enters. The password is then converted into a key that is used for authentication.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring MD5 Authentication on page 145
- Configuring SHA Authentication on page 145

authentication-sha

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
authentication-sha {
    authentication-password authentication-password;
}
```

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user *username*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the secure hash algorithm (SHA) as the authentication type for the SNMPv3 user.

NOTE: You can configure only one authentication type for each SNMPv3 user.

The remaining statement is explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring SHA Authentication on page 145

authorization

Supported Platforms ACX Series, EX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax authorization *authorization*;

Hierarchy Level [edit snmp community *community-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Set the access authorization for SNMP **Get**, **GetBulk**, **GetNext**, and **Set** requests.

Options ***authorization***—Access authorization level:

- **read-only**—Enable **Get**, **GetNext**, and **GetBulk** requests.
- **read-write**—Enable all requests, including **Set** requests. You must configure a view to enable **Set** requests.

Default: read-only

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the SNMP Community String on page 120

categories

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
categories {
  category;
}
```

Hierarchy Level [edit snmp trap-group *group-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Define the types of traps that are sent to the targets of the named trap group.

Default If you omit the **categories** statement, all trap types are included in trap notifications.

Options *category*—Name of a trap type: **authentication**, **chassis**, **configuration**, **link**, **remote-operations**, **rmon-alarm**, **routing**, **sonet-alarms**, **startup**, or **vrrp-events**.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Trap Groups on page 132

client-list

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
client-list client-list-name {
  ip-addresses;
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced in Junos OS Release 8.5.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for QFX Series switches.

Description Define a list of SNMP clients.

Options ***client-list-name***—Name of the client list.

ip-addresses—IP addresses of the SNMP clients to be added to the client list,

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Adding a Group of Clients to an SNMP Community on page 166

client-list-name

Supported Platforms ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax client-list-name *client-list-name*;

Hierarchy Level [edit snmp community *community-name*]

Release Information Statement introduced in Junos OS Release 8.5.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for FX Series switches.

Description Add a client list or prefix list to an SNMP community.

Options ***client-list-name***—Name of the client list or prefix list.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Adding a Group of Clients to an SNMP Community on page 166

clients

Supported Platforms ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
clients {
  address <restrict>;
}
```

Hierarchy Level [edit snmp community *community-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for FX Series switches.

Description Specify the IPv4 or IPv6 addresses of the SNMP client hosts that are authorized to use this community.

Default If you omit the **clients** statement, all SNMP clients using this community string are authorized to access the router.

Options ***address***—Address of an SNMP client that is authorized to access this router. You must specify an address, not a hostname. To specify more than one client, include multiple ***address*** options.

restrict—(Optional) Do not allow the specified SNMP client to access the router.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the SNMP Community String on page 120

commit-delay

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax commit-delay *seconds*;

Hierarchy Level [edit snmp nonvolatile]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure the timer for the SNMP **Set** reply and start of the commit.

Options ***seconds***—Delay between an affirmative SNMP **Set** reply and start of the commit.
Default: 5 seconds

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Commit Delay Timer on page 120

community

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
community community-name {
  authorization authorization;
  client-list-name client-list-name;
  clients {
    address restrict;
  }
  view view-name;
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Define an SNMP community. An SNMP community authorizes SNMP clients based on the source IP address of incoming SNMP request packets. A community also defines which MIB objects are available and the operations (read-only or read-write) allowed on those objects.

The SNMP client application specifies an SNMP community name in **Get**, **GetBulk**, **GetNext**, and **Set** SNMP requests.

Default If you omit the **community** statement, all SNMP requests are denied.

Options ***community-name***—Community string. If the name includes spaces, enclose it in quotation marks (" ").

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the SNMP Community String on page 120

community

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax community *community-name*;

Hierarchy Level [edit snmp rmon event *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The trap group that is used when generating a trap (if **eventType** is configured to send traps). If that trap group has the **rmon-alarm** trap category configured, a trap is sent to all the targets configured for that trap group. The community string in the trap matches the name of the trap group (and hence, the value of **eventCommunity**). If nothing is configured, traps are sent to each group with the **rmon-alarm** category set.

Options ***community-name***—Identifies the trap group that is used when generating a trap if the event is configured to send traps.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring an Event Entry and Its Attributes on page 228

community-name

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax community-name *community-name*;

Hierarchy Level [edit snmp v3 snmp-community *community-index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The community name defines an SNMP community. The SNMP community authorizes SNMPv1 or SNMPv2 clients. The access privileges associated with the configured security name define which MIB objects are available and the operations (notify, read, or write) allowed on those objects.

Options ***community-name***—Community string for an SNMPv1 or SNMPv2c community. If unconfigured, it is the same as the community index. If the name includes spaces, enclose it in quotation marks (" ").

NOTE: Community names must be unique. You cannot configure the same community name at the **[edit snmp community]** and **[edit snmp v3 snmp-community *community-index*]** hierarchy levels.

The community name at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level is encrypted and not displayed in the command-line interface (CLI).

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the SNMPv3 Community on page 167

contact

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax contact *contact*;

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Define the value of the MIB II **sysContact** object, which is the contact person for the managed system.

Options ***contact***—Name of the contact person. If the name includes spaces, enclose it in quotation marks (" ").

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the System Contact on a Device Running Junos OS on page 118

description

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax description *description*;

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Define the value of the MIB II **sysDescription** object, which is the description of the system being managed.

Options ***description***—System description. If the name includes spaces, enclose it in quotation marks (" ").

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the System Description on a Device Running Junos OS on page 119

description

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax description *description*;

Hierarchy Level [edit snmp rmon alarm *index*],
[edit snmp rmon event *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Text description of alarm or event.

Options ***description***—Text description of an alarm or event entry. If the description includes spaces, enclose it in quotation marks (" ").

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Description on page 225
- Configuring an Event Entry and Its Attributes on page 228

destination-port

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax destination-port *port-number*;

Hierarchy Level [edit snmp trap-group]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Assign a trap port number other than the default.

Default If you omit this statement, the default port is 162.

Options ***port-number***—SNMP trap port number.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Trap Groups on page 132

engine-id

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax

```
engine-id {
  (local engine-id-suffix | use-default-ip-address | use-mac-address);
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.1 for EX Series switches.

Description The local engine ID is defined as the administratively unique identifier of an SNMPv3 engine, and is used for identification, not for addressing. There are two parts of an engine ID: prefix and suffix. The prefix is formatted according to the specifications defined in RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*. You can configure the suffix here.

NOTE: SNMPv3 authentication and encryption keys are generated based on the associated passwords and the engine ID. If you configure or change the engine ID, you must commit the new engine ID before you configure SNMPv3 users. Otherwise the keys generated from the configured passwords are based on the previous engine ID.

For the engine ID, we recommend using the MAC address of the management port.

Options

local *engine-id-suffix*—Explicit setting for the engine ID suffix.

use-default-ip-address—The engine ID suffix is generated from the default IP address.

use-mac-address—The SNMP engine identifier is generated from the MAC address of the management interface on the router.

Default: use-default-ip-address

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Local Engine ID on page 156

enterprise-oid

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
enterprise-oid;
```

Hierarchy Level [edit snmp trap-options]

Release Information Statement introduced in Junos OS Release 10.0

Description Add the **snmpTrapEnterprise** object, which shows the association between an enterprise-specific trap and the organization that defined the trap, to standard SNMP traps. By default, the **snmpTrapEnterprise** object is added only to the enterprise-specific traps. When the **enterprise-oid** statement is included in the configuration, **snmpTrapEnterprise** is added to all the traps generated from the device.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Trap Options on page 128

event

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax
```
event index {
  community community-name;
  description description;
  type type;
}
```

Hierarchy Level [edit snmp rmon]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure RMON event entries.

Options *index*—Identifier for a specific event entry.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring an Event Entry and Its Attributes on page 228
- alarm (SNMP RMON) on page 256

falling-event-index

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax falling-event-index *index*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The index of the event entry that is used when a falling threshold is crossed. If this value is zero, no event is triggered.

Options *index*—Index of the event entry that is used when a falling threshold is crossed.

Range: 0 through 65,535

Default: 0

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Falling Event Index or Rising Event Index on page 225
- rising-event-index on page 304

falling-threshold

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax falling-threshold *percentage*;

Hierarchy Level [edit snmp]

Release Information Statement introduced in Junos OS Release 8.0.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The lower threshold is expressed as a percentage of the maximum possible value for the sampled variable. When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is 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. After a falling event is generated, another falling event cannot be generated until the sampled value rises above this threshold and reaches the **rising-threshold**.

Options *percentage*—The lower threshold for the alarm entry.

Range: 1 through 100

Default: 70 percent of the maximum possible value

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Falling Threshold or Rising Threshold on page 241
- rising-threshold on page 306

falling-threshold

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax falling-threshold *integer*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The lower threshold for the sampled variable. When the current sampled value is less than or equal to this threshold, and the value at the last sampling interval is 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 startup-alarm value is equal to falling-alarm value or rising-or-falling-alarm value. After a falling event is generated, another falling event cannot be generated until the sampled value rises above this threshold and reaches the **rising-threshold**.

Options *integer*—The lower threshold for the alarm entry.

Range: -2,147,483,648 through 2,147,483,647

Default: 20 percent less than **rising-threshold**

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Falling Threshold or Rising Threshold on page 225
- rising-threshold on page 305

falling-threshold-interval

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax falling-threshold-interval *seconds*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced in Junos OS Release 8.3.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Interval between samples when the rising threshold is crossed. Once the alarm crosses the falling threshold, the regular sampling interval is used.

Options ***seconds***—Time between samples, in seconds.
Range: 1 through 2,147,483,647 seconds
Default: 60 seconds

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Falling Threshold Interval on page 226
- interval on page 280

filter-duplicates

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax filter-duplicates;

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Filter duplicate **Get**, **GetNext**, or **GetBulk** SNMP requests.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Filtering Duplicate SNMP Requests on page 123

filter-interfaces

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
filter-interfaces {
  interfaces {
    all-internal-interfaces;
    interface 1;
    interface 2;
  }
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced in Junos OS Release 9.4.
Statement introduced in Junos OS Release 9.4 for EX Series Switches.

Description Filter out information related to specific interfaces from the output of SNMP **Get** and **GetNext** requests performed on interface-related MIBs.

Options

all-internal-interfaces—Filters out information from SNMP **Get** and **GetNext** requests for the specified interfaces.

interfaces—Specifies the interfaces to filter out from the output of SNMP **Get** and **GetNext** requests.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Filtering Interface Information Out of SNMP Get and GetNext Output on page 125

group (Configuring Group Name)

Supported Platforms ACX Series, EX Series, M Series, MX Series, OCX1100, PTX Series, QFabric System, QFX Series standalone switches, SRX Series, T Series

Syntax

```
group group-name {
  (default-context-prefix | context-prefix context-prefiix){
    security-model (any | usm | v1 | v2c) {
      security-level (authentication | none | privacy) {
        notify-view view-name;
        read-view view-name;
        write-view view-name;
      }
    }
  }
}
```

Hierarchy Level [edit snmp v3 vacm access]

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Assign the security name to a group, and specify the SNMPv3 context applicable to the group. The **default-context-prefix** statement, when included, adds all the contexts configured on the device to the group, whereas the **context-prefix *context-prefix*** statement enables you to specify a context and to add that particular context to the group.

(Not applicable to the QFX Series and OCX Series.) When the context prefix is specified as default (for example, **context-prefix default**), the context associated with the master routing instance is added to the group. To specify a routing instance that is part of a logical system, specify it as **logical system/routing instance**. For example, to specify routing instance ri1 in logical system ls1, include **context-prefix ls1/ri1**.

The remaining statements under this hierarchy are explained separately.

Options ***group-name***—SNMPv3 group name created for the SNMPv3 group.

Required Privilege Level

snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Group on page 150

group (Defining Access Privileges for an SNMPv3 Group)

Supported Platforms ACX Series, EX Series, M Series, MX Series, OCX1100, PTX Series, QFabric System, QFX Series standalone switches, SRX Series, T Series

Syntax
```
group group-name;
```

Hierarchy Level
```
[edit snmp v3 vacm security-to-group security-model (usm | v1 | v2c)
  security-name security-name]
```

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Define access privileges granted to a group.

Options *group-name*—Identifies a collection of SNMP security names that belong to the same access policy SNMP.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Group on page 154

health-monitor

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax
```
health-monitor {
  falling-threshold percentage;
  interval seconds;
  rising-threshold percentage;
}
```

Hierarchy Level
```
[edit snmp]
```

Release Information Statement introduced in Junos OS Release 8.0.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure health monitoring.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring Health Monitoring on Devices Running Junos OS on page 239

interface

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax interface [*interface-names*];

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the interfaces on which SNMP requests can be accepted.

Default If you omit this statement, SNMP requests entering the router or switch through any interface are accepted.

Options ***interface-names***—Names of one or more logical interfaces.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Interfaces on Which SNMP Requests Can Be Accepted on page 124

interval

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax interval *seconds*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Interval between samples.

Options ***seconds***—Time between samples, in seconds.
Range: 1 through 2,147,483,647 seconds
Default: 60 seconds

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Interval on page 226

interval

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax interval *seconds*;

Hierarchy Level [edit snmp health-monitor]

Release Information Statement introduced in Junos OS Release 8.0.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Interval between samples.

Options ***seconds***—Time between samples, in seconds.

Range: 1 through **2147483647** seconds

Default: **300** seconds

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Interval on page 241

local-engine

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
local-engine {
  user username {
    authentication-md5 {
      authentication-password authentication-password;
    }
    authentication-none;
    authentication-sha {
      authentication-password authentication-password;
    }
    privacy-aes128 {
      privacy-password privacy-password;
    }
    privacy-des {
      privacy-password privacy-password;
    }
    privacy-3des {
      privacy-password privacy-password;
    }
    privacy-none {
      privacy-password privacy-password;
    }
  }
}
```

Hierarchy Level [edit snmp v3 usm]

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure local engine information for the user-based security model (USM).

The remaining statements are explained separately.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Creating SNMPv3 Users on page 138

location

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax location *location*;

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Define the value of the MIB II **sysLocation** object, which is the physical location of the managed system.

Options ***location***—Location of the local system. You must enclose the name within quotation marks (" ").

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the System Location for a Device Running Junos OS on page 118

logical-system

Supported Platforms EX Series, LN Series, M120, MX240, PTX Series, SRX Series, SRX210, SRX3400, T1600

Syntax

```
logical-system logical-system-name {
  routing-instance routing-instance-name;
  source-address address;;
}
```

Hierarchy Level

[edit snmp community *community-name*],
[edit snmp trap-group],
[edit snmp trap-options]
[edit snmp v3target-address *target-address-name*]

Release Information

Statement introduced in Junos OS Release 9.3
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

NOTE: **The logical-system statement replaces the logical-router statement, and is backward-compatible with Junos OS Release 8.3 and later.**

Description

Specify a logical system name for SNMP v1 and v2c clients.

Include at the **[edit snmp trap-options]** hierarchy level to specify a logical-system address as the source address of an SNMP trap.

Include at the **[edit snmp v3 target-address]** hierarchy level to specify a logical-system name as the destination address for an SNMPv3 trap or inform.

Options

logical-system-name—Name of the logical system.

routing-instance ***routing-instance-name***—Statement to specify a routing instance associated with the logical system.

Required Privilege Level

snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180
- Configuring the Trap Target Address on page 161

logical-system-trap-filter

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
logical-system-trap-filter;
```

Hierarchy Level [edit snmp]

Release Information Statement introduced in Junos OS Release 8.4.

Description Restrict the routing instances from receiving traps that are not related to the logical system networks to which they belong.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Trap Support for Routing Instances on page 178

log-vital

Supported Platforms SRX Series

Syntax

```
log-vital {
  add <oid> {
    comment <comment>;
  }
  file-size;
  files;
  group {
    operating;
    idp;
    storage;
    cluster-counter;
    screen;
    spu;
  }
  interval;
  storage-limit;
}
```

Hierarchy Level [edit system]

Release Information Statement introduced in Junos OS Release 12.1X47-D15.

Description Configure vital log data.

Options **add***<oid>*—Specify the OID to be used to collect the raw data.

- **comment**—Specify the comment for the raw OID.

file-size—Specify the size of the current dump file.

Range: 1 MB to 100 MB.

Default: 5 MB for branch SRX Series devices and 10 MB for high-end SRX Series devices.

files—Specify the lifetime (number of days) for the dump file to be stored. The dump file is stored at **/var/log/vital/**.

Range: 1 to 30 days.

Default: 3 days.

group—Specify the pre-defined OID group to be used. Each group contains multiple OIDs within the same area. Once a group enabled, all OIDs in the group will be periodically collected and dumped.

- **operating**—This group includes state, temperature, current CPU utilization percentage, buffer utilization percentage, heap-utilization percentage, up time, average-load in the last 1 minute, 5 minutes, or 15 minutes, and buffer-pool utilization percentage in the control plane of each operating component in the system.

- **idp**—This group includes IDP data plane memory usage, IDP session usage and policies loaded number.
- **storage**—This group includes storage utilization of directory **/var/log**.
- **cluster-counter**—This group includes current total session number, total CPS, IPv4 CPS, IPv6 CPS, current total IPv4 session number, and current total IPv6 session number of both node 0 and node 1.
- **screen**—This group includes screen statistics of a specified zone.
- **spu**—This group includes CPU usage, memory usage, current flow session number, current CP session number, IPv4 session number, IPv6 session number, CP IPv4 session number, and CP IPv6 session number of the SPU.

interval—Specify the collection interval in minutes. The configuration takes effect immediately with new interval value.

Range: 1 to 1440 minutes.

Default: 10 minutes.

storage-limit—Specify the storage usage limit in percentage. If the current storage usage of the directory **/var/log/** is above the upper limit, collection is canceled but is tried next time.

Range: 1 to 100 percent.

Default: 80 percent.

Required Privilege Level

security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

Related Documentation

- show system log-vital on page 361

message-processing-model

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax message-processing-model (v1 | v2c | v3);

Hierarchy Level [edit snmp v3 target-parameters *target-parameter-name* parameters]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the message processing model to be used when generating SNMP notifications.

Options **v1**—SNMPv1 message process model.

v2c—SNMPv2c message process model.

v3—SNMPv3 message process model.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Message Processing Model on page 164

name

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax name *name*;

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Set the system name from the command-line interface.

Options ***name***—System name override.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the System Name on page 119

nonvolatile

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
nonvolatile {
    commit-delay seconds;
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
The commit-delay statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure options for SNMP **Set** requests.

The statement is explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Commit Delay Timer on page 120
- commit-delay on page 265

notify

Supported Platforms ACX Series, EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax

```
notify name {
  tag tag-name;
  type (trap | inform);
}
```

Hierarchy Level [edit snmp v3]

Release Information
Statement introduced before Junos OS Release 7.4.
type inform option added in Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Select management targets for SNMPv3 notifications as well as the type of notifications. Notifications can be either traps or informs.

Options

name—Name assigned to the notification.

tag-name—Notifications are sent to all targets configured with this tag.

type—Notification type is **trap** or **inform**. Traps are unconfirmed notifications. Informs are confirmed notifications.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Inform Notification Type and Target Address on page 170
- Configuring the SNMPv3 Trap Notification on page 159

notify-filter (Applying to the Management Target)

Supported Platforms ACX Series, EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
notify-filter profile-name;
```

Hierarchy Level [edit snmp v3 target-parameters *target-parameters-name*]

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Specify the notify filter to be used by a specific set of target parameters.

Options ***profile-name***—Name of the notify filter to apply to notifications.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Applying the Trap Notification Filter on page 164

notify-filter (Configuring the Profile Name)

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax
```
notify-filter profile-name {
  oid oid (include | exclude);
}
```

Hierarchy Level [edit snmp v3]

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Specify a group of MIB objects for which you define access. The notify filter limits the type of traps or informs sent to the network management system.

Options ***profile-name***—Name assigned to the notify filter.

The remaining statement is explained separately.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Trap Notification Filter on page 135
- oid on page 294

notify-view

Supported Platforms ACX Series, EX Series, M Series, MX Series, OCX1100, PTX Series, QFabric System, QFX Series standalone switches, SRX Series, T Series

Syntax notify-view *view-name*;

Hierarchy Level [edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Associate the notify view with a community (for SNMPv1 or SNMPv2c clients) or a group name (for SNMPv3 clients).

Options ***view-name***—Name of the view to which the SNMP user group has access.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring MIB Views on page 126
- Configuring the Notify View on page 151

oid

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax oid *object-identifier* (exclude | include);

Hierarchy Level [edit snmp view *view-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Specify an object identifier (OID) used to represent a subtree of MIB objects.

Options

exclude—Exclude the subtree of MIB objects represented by the specified OID.

include—Include the subtree of MIB objects represented by the specified OID.

object-identifier—OID used to represent a subtree of MIB objects. All MIB objects represented by this statement have the specified OID as a prefix. You can specify the OID using either a sequence of dotted integers or a subtree name.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring MIB Views on page 126

oid

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax oid *oid* (include | exclude);

Hierarchy Level [edit snmp v3 notify-filter *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Specify an object identifier (OID) used to represent a subtree of MIB objects. This OID is a prefix that the represented MIB objects have in common.

Options

exclude—Exclude the subtree of MIB objects represented by the specified OID.

include—Include the subtree of MIB objects represented by the specified OID.

oid—Object identifier used to represent a subtree of MIB objects. All MIB objects represented by this statement have the specified OID as a prefix. You can specify the OID using either a sequence of dotted integers or a subtree name.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Trap Notification Filter on page 135

parameters

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
parameters {
  message-processing-model (v1 | v2c | v3);
  security-level (none | authentication | privacy);
  security-model (usm | v1 | v2c);
  security-name security-name;
}
```

Hierarchy Level [edit snmp v3 target-parameters *target-parameters-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure a set of target parameters for message processing and security.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Defining and Configuring the Trap Target Parameters on page 163

port

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax port *port-number*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure a UDP port number for an SNMP target.

Default If you omit this statement, the default port is 162.

Options ***port-number***—Port number for the SNMP target.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Port on page 162

privacy-3des

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
privacy-3des {
    privacy-password privacy-password;
}
```

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user *username*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the triple Data Encryption Standard (3DES) as the privacy type for the SNMPv3 user.

Options **privacy-password *privacy-password***—Password that a user enters. The password is then converted into a key that is used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Encryption Type on page 146

privacy-aes128

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
privacy-aes128 {
    privacy-password privacy-password;
}
```

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user *username*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the Advanced Encryption Standard encryption algorithm (CFB128-AES-128 Privacy Protocol) for the SNMPv3 user.

Options **privacy-password *privacy-password***—Password that a user enters. The password is then converted into a key that is used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Encryption Type on page 146

privacy-des

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
privacy-des {
    privacy-password privacy-password;
}
```

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user *username*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the Data Encryption Standard (DES) as the privacy type for the SNMPv3 user.

Options **privacy-password *privacy-password***—Password that a user enters. The password is then converted into a key that is used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Encryption Type on page 146

privacy-none

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax
```
privacy-none;
```

Hierarchy Level [edit snmp v3 usm local-engine user *username*],
[edit snmp v3 usm remote-engine *engine-id* user *username*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure that no encryption be used for the SNMPv3 user.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Encryption Type on page 146

privacy-password

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax privacy-password *privacy-password*;

Hierarchy Level [edit snmp v3 usm local-engine user *username* privacy-3des],
[edit snmp v3 usm local-engine user *username* privacy-aes128],
[edit snmp v3 usm local-engine user *username* privacy-des],
[edit snmp v3 usm remote-engine *engine-id* user *username* privacy-3des],
[edit snmp v3 usm remote-engine *engine-id* user *username* privacy-aes128],
[edit snmp v3 usm remote-engine *engine-id* user *username* privacy-des]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure a privacy password for the SNMPv3 user.

Options ***privacy-password***—Password that a user enters. The password is then converted into a key that is used for encryption.

SNMPv3 has special requirements when you create plain-text passwords on a router or switch:

- The password must be at least eight characters long.
- The password can include alphabetic, numeric, and special characters, but it cannot include control characters.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Encryption Type on page 146

read-view

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax
```
read-view view-name;
```

Hierarchy Level
```
[edit snmp v3 vacm access group group-name (default-context-prefix | context-prefix
  context-prefix) security-model (any | usm | v1 | v2c) security-level (authentication | none |
  privacy)]
```

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Associate the read-only view with a community (for SNMPv1 or SNMPv2c clients) or a group name (for SNMPv3 clients).

Options ***view-name***—The name of the view to which the SNMP user group has access.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Read View on page 152
- Configuring MIB Views on page 126

remote-engine

Supported Platforms M Series, MX Series, PTX Series, QFX Series, SRX Series, T Series

Syntax

```
remote-engine engine-id {
  user username {
    authentication-md5 {
      authentication-password authentication-password;
    }
    authentication-none;
    authentication-sha {
      authentication-password authentication-password;
    }
    privacy-aes128 {
      privacy-password privacy-password;
    }
    privacy-des {
      privacy-password privacy-password;
    }
    privacy-3des {
      privacy-password privacy-password;
    }
    privacy-none {
      privacy-password privacy-password;
    }
  }
}
```

Hierarchy Level [edit snmp v3 usm]

Release Information
Statement introduced in Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the remote engine information for the user-based security model (USM). To send inform messages to an SNMPv3 user on a remote device, you must configure the engine identifier for the SNMP agent on the remote device where the user resides.

Options
engine-id—Engine identifier. Used to compute the security digest for authenticating and encrypting packets sent to a user on the remote host.

The remaining statements are explained separately.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Remote Engine and Remote User on page 172

request-type

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax request-type (get-next-request | get-request | walk-request);

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced in Junos OS Release 8.3.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Extend monitoring to a specific SNMP object instance (**get-request**), or extend monitoring to all object instances belonging to a MIB branch (**walk-request**), or extend monitoring to the next object instance after the instance specified in the configuration (**get-next-request**).

Options

get-next-request—Performs an SNMP get next request.

get-request—Performs an SNMP get request.

walk-request—Performs an SNMP walk request.

Default: walk-request

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Request Type on page 226
- variable on page 341

retry-count

Supported Platforms LN Series

Syntax retry-count *number*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced in Junos OS Release 7.4.

Description Configure the retry count for SNMP informs.

Options *number*—Maximum number of times the inform is transmitted if no acknowledgment is received. If no acknowledgment is received after the inform is transmitted the maximum number of times, the inform message is discarded.

Default: 3 times

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring SNMP Informs on page 157
- timeout on page 328

rising-event-index

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax rising-event-index *index*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Index of the event entry that is used when a rising threshold is crossed. If this value is zero, no event is triggered.

Options *index*—Index of the event entry that is used when a rising threshold is crossed.

Range: 0 through 65,535

Default: 0

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Falling Event Index or Rising Event Index on page 225
- falling-event-index on page 273

rising-threshold

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax rising-threshold *integer*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Upper threshold for the sampled variable. When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is 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 startup alarm value is equal to the falling alarm or rising or falling alarm value. After a rising event is generated, another rising event cannot be generated until the sampled value falls below this threshold and reaches the falling threshold.

Options *integer*—The lower threshold for the alarm entry.
Range: –2,147,483,648 through 2,147,483,647

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Falling Threshold or Rising Threshold on page 225
- falling-threshold on page 275

rising-threshold

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax rising-threshold *percentage*;

Hierarchy Level [edit snmp]

Release Information Statement introduced in Junos OS Release 8.0.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The upper threshold is expressed as a percentage of the maximum possible value for the sampled variable. When the current sampled value is greater than or equal to this threshold, and the value at the last sampling interval is 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. After a rising event is generated, another rising event cannot be generated until the sampled value falls below this threshold and reaches the **falling-threshold**.

Options *percentage*—The lower threshold for the alarm entry.

Range: 1 through 100

Default: 80 percent of the maximum possible value

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- falling-threshold on page 274
- Configuring the Falling Threshold or Rising Threshold on page 241

rmon

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax rmon { ... }

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure Remote Monitoring.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring an Alarm Entry and Its Attributes on page 224

routing-engine (SNMP Resource Level)

Supported Platforms LN Series, SRX100, SRX110, SRX210, SRX220, SRX240, SRX550, SRX650, vSRX

Syntax

```
routing-engine {
  resource <cpu | memory | open-files-count | process-count | storage | temperature> ;
  {
    interval <interval in secs>;
    moderate-threshold <percentage level>;
    high-threshold <percentage level>;
    critical-threshold <percentage level>;
    action <monitor | prevent | recover>;
  }
}
```

Hierarchy Level [edit snmp health-monitor routing-engine]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. Statement modified in Junos OS Release 15.1x49-D10.

Description Override the global configuration for a resource.

Options

- **interval**—Monitoring interval in seconds.

  Default: 300 seconds
- **moderate-threshold**—Percentage of moderate threshold level resource utilization.

  Default: 70 percent.
- **high-threshold** —Percentage of high-threshold level resource utilization.

  Default: 80 percent.
- **critical-threshold** —Percentage of critical threshold level resource utilization.

  Default: 90 percent.
- **action**—Enable action for all resources.

  Default: If action is not enabled, the default action is prevent.

WARNING: If the system health management action for an affected resource is configured to recover, then certain instrusive operations necessary for preventing system breakdown are taken. Instrusive operations can include restarting or terminating processes, deleting files, and so on. Such action information is logged in the system health management history and system log.

Required Privilege Level security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

routing-engine (SNMP Global Level)

Supported Platforms LN Series, SRX100, SRX110, SRX210, SRX220, SRX240, SRX550, SRX650, vSRX

Syntax

```
routing-engine
  {
    interval <interval in secs>;
    moderate-threshold <percentage level>;
    high-threshold <percentage level>;
    critical-threshold <percentage level>;
    traceoptions;
    action <monitor | prevent | recover>;
  }
```

Hierarchy Level [edit snmp health-monitor routing-engine]

Release Information Statement introduced in Junos OS Release 12.1X44-D10. Statement modified in Junos OS Release 12.1X45-D10.

Description Enable the system health management feature to use the specified parameters.

Options

- **interval**—Monitoring interval in seconds.

  Default: 300 seconds

- **moderate-threshold**—Percentage of moderate threshold level resource utilization.

  Default: 70 percent.

- **high-threshold** —Percentage of high-threshold level resource utilization.

  Default: 80 percent.

- **critical-threshold** —Percentage of critical threshold level resource utilization.

  Default: 90 percent.

- **traceoptions**—Enable tracing of system health monitoring daemon.

- **action**—Enable action for all resources.

  Default: If action is not enabled, the default is prevent.

WARNING: If the system health management action for an affected resource is configured to recover, then certain instrusive operations necessary for preventing system breakdown are taken. Instrusive operations can include restarting or terminating processes, deleting files, and so on. Such action information is logged in the system health management history and system log.

Required Privilege Level security—To view this statement in the configuration.
security-control—To add this statement to the configuration.

routing-instance

Supported Platforms ACX Series, EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
routing-instance routing-instance-name;
```

Hierarchy Level
```
[edit snmp community community-name],
[edit snmp community community-name logical-system logical-system-name],
[edit snmp trap-group group]
```

Release Information Statement introduced in Junos OS Release 8.3.
Added to the **[edit snmp community *community-name*]** hierarchy level in Junos OS Release 8.4.
Added to the **[edit snmp community *community-name* logical-system *logical-system-name*]** hierarchy level in Junos OS Release 9.1.
Statement introduced in Junos OS Release 9.1 for EX Series switches.

Description Specify a routing instance for SNMPv1 and SNMPv2 trap targets. All targets configured in the trap group use this routing instance.

If the routing instance is defined within a logical system, include the **logical-system *logical-system-name*** statement at the **[edit snmp community *community-name*]** hierarchy level and specify the **routing-instance** statement under the **[edit snmp community *community-name* logical-system *logical system-name*]** hierarchy level.

Options ***routing-instance-name***—Name of the routing instance.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Trap Groups on page 132
- Configuring the Source Address for SNMP Traps on page 129
- Specifying a Routing Instance in an SNMPv1 or SNMPv2c Community on page 180

routing-instance

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax routing-instance *routing-instance-name*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced in Junos OS Release 8.3.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Specify a routing instance for an SNMPv3 trap target.

Options *routing-instance-name*—Name of the routing instance.

To configure a routing instance within a logical system, specify the logical system name followed by the routing instance name. Use a slash (/) to separate the two names (for example, **test-ls/test-ri**). To configure the default routing instance on a logical system, specify the logical system name followed by **default** (for example, **test-ls/default**).

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Trap Target Address on page 161

routing-instance-access

Supported Platforms LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
[edit snmp]
  routing-instance-access {
    access-list {
      routing-instance;
      routing-instance restrict;
    }
  }
```

Hierarchy Level [edit snmp]

Release Information Statement introduced in Junos OS Release 8.4.

Description Enable SNMP managers in routing instances other than the default routing instance to access SNMP information. For information about the **access-list** option, see access-list.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Enabling SNMP Access over Routing Instances on page 180

sample-type

Supported Platforms EX Series, LN Series, M Series, MX Series, SRX Series, T Series

Syntax sample-type (absolute-value | delta-value);

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Method of sampling the selected variable.

Options

absolute-value—Actual value of the selected variable is used when comparing against the thresholds.

delta-value—Difference between samples of the selected variable is used when comparing against the thresholds.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Sample Type on page 227

security-level (Defining Access Privileges)

Supported Platforms EX Series, LN Series, M Series, MX Series, QFX Series, T Series

Syntax

```
security-level (authentication | none | privacy) {
  notify-view view-name;
  read-view view-name;
  write-view view-name;
}
```

Hierarchy Level [edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c)]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Define the security level used for access privileges.

Default none

Options

authentication—Provide authentication but no encryption.

none—No authentication and no encryption.

privacy—Provide authentication and encryption.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Level on page 150

security-level (Generating SNMP Notifications)

Supported Platforms EX Series, LN Series, M Series, MX Series, QFX Series, T Series

Syntax security-level (authentication | none | privacy);

Hierarchy Level [edit snmp v3 target-parameters *target-parameters-name* parameters]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the security level to use when generating SNMP notifications.

Default none

Options **authentication**—Provide authentication but no encryption.

none—No authentication and no encryption.

privacy—Provide authentication and encryption.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Level on page 165

security-model (Access Privileges)

Supported Platforms EX Series, LN Series, M Series, MX Series, QFX Series, T Series

Syntax security-model (usm | v1 | v2c);

Hierarchy Level [edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*)]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the security model for an SNMPv3 group. The security model is used to determine access privileges for the group.

Options

usm—SNMPv3 security model.

v1—SNMPv1 security model.

v2c—SNMPv2c security model.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Model on page 150

security-model (Group)

Supported Platforms EX Series, LN Series, M Series, MX Series, T Series

Syntax

```
security-model (usm | v1 | v2c) {
  security-name security-name {
    group group-name;
  }
}
```

Hierarchy Level [edit snmp v3 vacm security-to-group]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Define a security model for a group.

Options

usm—SNMPv3 security model.

v1—SNMPv1 security model.

v2c—SNMPv2c security model.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Model on page 154

security-model (SNMP Notifications)

Supported Platforms EX Series, LN Series, M Series, MX Series, QFX Series, T Series

Syntax security-model (usm | v1 | v2c);

Hierarchy Level [edit snmp v3 target-parameters *target-parameters-name* parameters]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the security model for an SNMPv3 group. The security model is used for SNMP notifications.

Options

usm—SNMPv3 security model.

v1—SNMPv1 security model.

v2c—SNMPv2c security model.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Model on page 165

security-name (Community String)

Supported Platforms ACX Series, EX Series, M Series, MX Series, QFX Series, T Series

Syntax security-name *security-name*;

Hierarchy Level [edit snmp v3 snmp-community *community-index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Associate a community string with the security name of a user. The community string, which is used for SNMPv1 and SNMPv2c clients in an SNMPv3 system, is configured at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level.

Options ***security-name***—Name that is used for messaging security and user access control.

NOTE: The security name must match the configured security name at the **[edit snmp v3 target-parameters *target-parameters-name* parameters]** hierarchy level when you configure traps or informs.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Names on page 169

security-name (Security Group)

Supported Platforms EX Series, LN Series, M Series, MX Series, T Series

Syntax

```
security-name security-name {
    group group-name;
}
```

Hierarchy Level [edit snmp v3 vacm security-to-group security-model (usm | v1 | v2c)]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Associate a group or a community string with a configured security group.

Options *security-name*—Username configured at the **[edit snmp v3 usm local-engine user *username*]** hierarchy level. For SNMPv1 and SNMPv2c, the security name is the community string configured at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Assigning Security Names to Groups on page 154

security-name (SNMP Notifications)

Supported Platforms EX Series, LN Series, M Series, MX Series, QFX Series, T Series

Syntax security-name *security-name*;

Hierarchy Level [edit snmp v3 target-parameters *target-parameters-name* parameters]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the security name used when generating SNMP notifications.

Options ***security-name***—If the SNMPv3 USM security model is used, identify the user when generating the SNMP notification. If the v1 or v2c security models are used, identify the SNMP community used when generating the notification.

NOTE: The access privileges for the group associated with this security name must allow this notification to be sent.

If you are using the v1 or v2 security models, the security name at the **[edit snmp v3 vacm security-to-group]** hierarchy level must match the security name at the **[edit snmp v3 snmp-community *community-index*]** hierarchy level.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Security Name on page 165

security-to-group

Supported Platforms EX Series, LN Series, M Series, MX Series, QFX Series, T Series

Syntax

```
security-to-group {
  security-model (usm | v1 | v2c) {
    group group-name;
    security-name security-name;
  }
}
```

Hierarchy Level [edit snmp v3 vacm]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the group to which a specific SNMPv3 security name belongs. The security name is used for messaging security.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Assigning Security Model and Security Name to a Group on page 153

snmp

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
snmp { ... }
```

Hierarchy Level [edit]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure SNMP.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring SNMP on a Device Running Junos OS on page 115

source-address

Supported Platforms ACX Series, EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax source-address *address*;

Hierarchy Level [edit snmp trap-options]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Set the source address of every SNMP trap packet sent by this router to a single address regardless of the outgoing interface. If the source address is not specified, the default is to use the address of the outgoing interface as the source address.

Options ***address***—Source address of SNMP traps. You can configure the source address of trap packets two ways: **lo0** or a valid IPv4 address configured on one of the router interfaces. The value **lo0** indicates that the source address of all SNMP trap packets is set to the lowest loopback address configured at interface **lo0**.

Default: Disabled. (The source address is the address of the outgoing interface.)

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Source Address for SNMP Traps on page 129

snmp-community

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax

```
snmp-community community-index {
  community-name community-name;
  security-name security-name;
  tag tag-name;
}
```

Hierarchy Level [edit snmp v3]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure the SNMP community.

Options ***community-index***—(Optional) String that identifies an SNMP community.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the SNMPv3 Community on page 167

startup-alarm

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax startup-alarm (falling-alarm | rising-alarm | rising-or-falling-alarm);

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The alarm that can be sent upon entry startup.

Options

falling-alarm—Generated if the first sample after the alarm entry becomes active is less than or equal to the falling threshold.

rising-alarm—Generated if the first sample after the alarm entry becomes active is greater than or equal to the rising threshold.

rising-or-falling-alarm—Generated if the first sample after the alarm entry becomes active satisfies either of the corresponding thresholds.

Default: rising-or-falling-alarm

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Startup Alarm on page 227

syslog-subtag

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax syslog-subtag *syslog-subtag*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced in Junos OS Release 8.5.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Add a tag to the system log message.

Options **syslog-subtag *syslog-subtag***—Tag of not more than 80 uppercase characters to be added to syslog messages.

Default: None

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the System Log Tag on page 227

tag

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax tag *tag-name*;

Hierarchy Level [edit snmp v3 notify *name*],
[edit snmp v3 snmp-community *community-index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure a set of targets to receive traps or informs (for IPv4 packets only).

Options ***tag-name***—Identifies the address of managers that are allowed to use a community string.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Tag on page 169
- Configuring the SNMPv3 Trap Notification on page 159

tag-list

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax tag-list *tag-list*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure an SNMP tag list used to select target addresses.

Options ***tag-list***—Define sets of target addresses (tags). To specify more than one tag, specify the tag names as a space-separated list enclosed within double quotes.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Trap Target Address on page 162

target-address

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax

```
target-address target-address-name {
  address address;
  address-mask address-mask;
  logical-system logical-system;
  port port-number;
  retry-count number;
  routing-instance instance;
  tag-list tag-list;
  target-parameters target-parameters-name;
  timeout seconds;
}
```

Hierarchy Level [edit snmp v3]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure the address of an SNMP management application and the parameters to be used in sending notifications.

Options *target-address-name*—String that identifies the target address.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the Trap Target Address on page 161

target-parameters

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax At the **[edit snmp v3]** hierarchy level:

```
target-parameters target-parameters-name {
  profile-name;
  parameters {
    message-processing-model (v1 | v2c | V3);
    security-level (authentication | none | privacy);
    security-model (usm | v1 | v2c);
    security-name security-name;
  }
}
```

At the **[edit snmp v3 target-address *target-address-name*]** hierarchy level:

```
target-parameters target-parameters-name;
```

Hierarchy Level
[edit snmp v3]
[edit snmp v3 target-address *target-address-name*]

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the message processing and security parameters for sending notifications to a particular management target. The target parameters are configured at the **[edit snmp v3]** hierarchy level. The remaining statements at this level are explained separately.

Then apply the target parameters configured at the **[edit snmp v3 target-parameters *target-parameters-name*]** hierarchy level to the target address configuration at the **[edit snmp v3]** hierarchy level.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Defining and Configuring the Trap Target Parameters on page 163
- Applying Target Parameters on page 163

targets

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax
```
targets {
  address;
}
```

Hierarchy Level [edit snmp trap-group *group-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure one or more systems to receive SNMP traps.

Options ***address***—IPv4 or IPv6 address of the system to receive traps. You must specify an address, not a hostname.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Trap Groups on page 132

timeout

Supported Platforms LN Series

Syntax timeout *seconds*;

Hierarchy Level [edit snmp v3 target-address *target-address-name*]

Release Information Statement introduced in Junos OS Release 7.4.

Description Configure the timeout period (in seconds) for SNMP informs.

Options ***seconds***—Number of seconds to wait for an inform acknowledgment. If no acknowledgment is received within the timeout period, the inform is retransmitted.
Default: 15

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Informs on page 157
- retry-count on page 304

traceoptions (SNMP)

Supported Platforms ACX Series, EX Series, M Series, MX Series, PTX Series, SRX210, SRX3400, vSRX, T Series

Syntax

```
traceoptions {
  file filename <files number> <match regular-expression> <size size> <world-readable |
    no-world-readable>;
  flag flag;
  no-remote-trace;
}
```

Hierarchy Level [edit snmp]

Release Information
Statement introduced before Junos OS Release 7.4.
file *filename* option added in Junos OS Release 8.1.
world-readable | no-world-readable option added in Junos OS Release 8.1.
match *regular-expression* option added in Junos OS Release 8.1.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description The output of the tracing operations is placed into log files in the **/var/log** directory. Each log file is named after the SNMP agent that generates it. Currently, the following logs are created in the **/var/log** directory when the **traceoptions** statement is used:

- chassisd
- craftd
- ilmid
- mib2d
- rmopd
- serviced
- snmpd

Options

file *filename*—By default, the name of the log file that records trace output is the name of the process being traced (for example, **mib2d** or **snmpd**). Use this option to specify another name.

files *number*—(Optional) Maximum number of trace files per SNMP subagent. When a trace file (for example, **snmpd**) reaches its maximum size, it is archived by being renamed to **snmpd.0**. The previous **snmpd.1** is renamed to **snmpd.2**, and so on. The oldest archived file is deleted.

Range: 2 through 1000 files

Default: 10 files

flag *flag*—Tracing operation to perform. To specify more than one tracing operation, include multiple **flag** statements:

- **all**—Log all SNMP events.

- **general**—Log general events.
- **interface-stats**—Log physical and logical interface statistics.
- **nonvolatile-sets**—Log nonvolatile SNMP set request handling.
- **pdu**—Log SNMP request and response packets.
- **protocol-timeouts**—Log SNMP response timeouts.
- **routing-socket**—Log routing socket calls.
- **subagent**—Log subagent restarts.
- **timer**—Log internally generated events.
- **varbind-error**—Log variable binding errors.

match *regular-expression*—(Optional) Refine the output to include lines that contain the regular expression.

size *size*—(Optional) Maximum size, in kilobytes (KB), of each trace file before it is closed and archived.

Range: 10 KB through 1 GB

Default: 1000 KB

world-readable | no-world-readable—(Optional) By default, log files can be accessed only by the user who configures the tracing operation. The **world-readable** option enables any user to read the file. To explicitly set the default behavior, use the **no-world-readable** option.

Required Privilege Level

snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Tracing SNMP Activity on a Device Running Junos OS on page 197

trap-group

Supported Platforms ACX Series, EX Series, M Series, MX Series, OCX1100, PTX Series, QFabric System, QFX Series, SRX Series, T Series

Syntax

```
trap-group group-name {
  categories {
    category;
  }
  destination-port port-number;
  routing-instance instance;
  targets {
    address;
  }
  version (all | v1 | v2);
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 14.1X53-D20 for OCX Series switches.

Description Create a named group of hosts to receive the specified trap notifications. The name of the trap group is embedded in SNMP trap notification packets as one variable binding (varbind) known as the community name. At least one trap group must be configured for SNMP traps to be sent.

Options *group-name*—Name of the trap group. If the name includes spaces, enclose it in quotation marks (" ").

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring SNMP Trap Groups on page 132

trap-options

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
trap-options {
  agent-address outgoing-interface;
  source-address address;
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Using SNMP trap options, you can set the source address of every SNMP trap packet sent by the router or switch to a single address, regardless of the outgoing interface. In addition, you can set the agent address of each SNMPv1 trap. For more information about the contents of SNMPv1 traps, see RFC 1157.

The remaining statements are explained separately.

Default Disabled

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring SNMP Trap Options on page 128

type

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax
```
type (inform | trap);
```

Hierarchy Level [edit snmp v3 notify *name*]

Release Information
Statement introduced before Junos OS Release 7.4.
inform option added in Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description Configure the type of SNMP notification.

Options
inform—Defines the type of notification as an inform. SNMP informs are confirmed notifications.

trap—Defines the type of notification as a trap. SNMP traps are unconfirmed notifications.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Informs on page 157
- Configuring the SNMPv3 Trap Notification on page 159

type

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax type *type*;

Hierarchy Level [edit snmp rmon event *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Type of notification generated when a threshold is crossed.

Options *type*—Type of notification:

- **log**—Add an entry to **logTable**.
- **log-and-trap**—Send an SNMP trap and make a log entry.
- **none**—No notifications are sent.
- **snmptrap**—Send an SNMP trap.

Default: log-and-trap

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring an Event Entry and Its Attributes on page 228

user

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax user *username*;

Hierarchy Level [edit snmp v3 usm local-engine],
[edit snmp v3 usm remote-engine *engine-id*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.
Statement introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Specify a user associated with an SNMPv3 group on a local or remote SNMP engine.

Options ***username***—SNMPv3 user-based security model (USM) username.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Creating SNMPv3 Users on page 138

usm

Supported Platforms LN Series, M Series, MX Series, PTX Series, QFX Series, T Series

Syntax

```
usm {
  local-engine {
    user username {
      authentication-md5 {
        authentication-password authentication-password;
      }
      authentication-none;
      authentication-sha {
        authentication-password authentication-password;
      }
      privacy-aes128 {
        privacy-password privacy-password;
      }
      privacy-des {
        privacy-password privacy-password;
      }
      privacy-3des {
        privacy-password privacy-password;
      }
      privacy-none {
        privacy-password privacy-password;
      }
    }
    remote-engine engine-id {
      user username {
        authentication-md5 {
          authentication-password authentication-password;
        }
        authentication-none;
        authentication-sha {
          authentication-password authentication-password;
        }
        privacy-aes128 {
          privacy-password privacy-password;
        }
        privacy-des {
          privacy-password privacy-password;
        }
        privacy-3des {
          privacy-password privacy-password;
        }
        privacy-none {
          privacy-password privacy-password;
        }
      }
    }
  }
}
```

Hierarchy Level [edit snmp v3]

Release Information
Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series.

Description
Configure user-based security model (USM) information.

The remaining statements are explained separately.

Required Privilege Level
snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Creating SNMPv3 Users on page 138
- Configuring the Remote Engine and Remote User on page 172

v3

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax

```
v3 {
  notify name {
    tag tag-name;
    type trap;
  }
  notify-filter profile-name {
    oid object-identifier (include | exclude);
  }
  snmp-community community-index {
    community-name community-name;
    security-name security-name;
    tag tag-name;
  }
  target-address target-address-name {
    address address;
    address-mask address-mask;
    logical-system logical-system;
    port port-number;
    retry-count number;
    routing-instance instance;
    tag-list tag-list;
    target-parameters target-parameters-name;
    timeout seconds;
  }
  target-parameters target-parameters-name {
    notify-filter profile-name;
    parameters {
      message-processing-model (v1 | v2c | V3);
      security-level (authentication | none | privacy);
      security-model (usm | v1 | v2c);
      security-name security-name;
    }
  }
  usm {
    local-engine {
      user username {
        authentication-md5 {
          authentication-password authentication-password;
        }
        authentication-sha {
          authentication-password authentication-password;
        }
        authentication-none;
        privacy-aes128 {
          privacy-password privacy-password;
        }
        privacy-des {
          privacy-password privacy-password;
        }
        privacy-des {
          privacy-password privacy-password;
```

```
            }
            privacy-none;
          }
        }
        remote-engine engine-id {
          user username {
            authentication-md5 {
              authentication-password authentication-password;
            }
            authentication-sha {
              authentication-password authentication-password;
            }
            authentication-none;
            privacy-aes128 {
              privacy-password privacy-password;
            }
            privacy-des {
              privacy-password privacy-password;
            }
            privacy-3des {
              privacy-password privacy-password;
            }
            privacy-none {
              privacy-password privacy-password;
            }
          }
        }
      }
      vacm {
        access {
          group group-name {
            (default-context-prefix | context-prefix context-prefix){
              security-model (any | usm | v1 | v2c) {
                security-level (authentication | none | privacy) {
                  notify-view view-name;
                  read-view view-name;
                  write-view view-name;
                }
              }
            }
          }
        }
        security-to-group {
          security-model (usm | v1 | v2c) {
            security-name security-name {
              group group-name;
            }
          }
        }
      }
    }
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure SNMPv3.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Minimum SNMPv3 Configuration on a Device Running Junos OS on page 143

vacm

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax

```
vacm {
  access {
    group group-name {
      (default-context-prefix | context-prefix context-prefix){
        security-model (any | usm | v1 | v2c) {
          security-level (authentication | none | privacy) {
            notify-view view-name;
            read-view view-name;
            write-view view-name;
          }
        }
      }
    }
  }
  security-to-group {
    security-model (usm | v1 | v2c);
    security-name security-name {
      group group-name;
    }
  }
}
```

Hierarchy Level [edit snmp v3]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Configure view-based access control model (VACM) information.

The remaining statements are explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Defining Access Privileges for an SNMP Group on page 148

variable

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, T Series

Syntax variable *oid-variable*;

Hierarchy Level [edit snmp rmon alarm *index*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Object identifier (OID) of MIB variable to be monitored.

Options ***oid-variable***—OID of the MIB variable that is being monitored. The OID can be a dotted decimal (for example, **1.3.6.1.2.1.2.1.2.2.1.10.1**). Alternatively, use the MIB object name (for example, **ifInOctets.1**).

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring the Variable on page 228

version

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax version (all | v1 | v2);

Hierarchy Level [edit snmp trap-group *group-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Specify the version number of SNMP traps.

Default **all**—Send an SNMPv1 and SNMPv2 trap for every trap condition.

Options **all**—Send an SNMPv1 and SNMPv2 trap for every trap condition.

v1—Send SNMPv1 traps only.

v2—Send SNMPv2 traps only.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring SNMP Trap Groups on page 132

view (Associating a MIB View with a Community)

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax view *view-name*;

Hierarchy Level [edit snmp community *community-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Associate a view with a community. A view represents a group of MIB objects.

Options *view-name*—Name of the view. You must use a view name already configured in the **view** statement at the **[edit snmp]** hierarchy level.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring the SNMP Community String on page 120

view (Configuring a MIB View)

Supported Platforms EX Series, LN Series, M Series, MX Series, PTX Series, SRX Series, T Series

Syntax

```
view view-name {
  oid object-identifier (include | exclude);
}
```

Hierarchy Level [edit snmp]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.

Description Define a MIB view. A MIB view identifies a group of MIB objects. Each MIB object in a view has a common OID prefix. Each object identifier represents a subtree of the MIB object hierarchy. The **view** statement uses a view to specify a group of MIB objects on which to define access. To enable a view, you must associate the view with a community by including the **view** statement at the **[edit snmp community *community-name*]** hierarchy level.

NOTE: To remove an OID completely, use the delete view all oid oid-number command but omit the include parameter.

Options *view-name*—Name of the view.

The remaining statement is explained separately.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation

- Configuring MIB Views on page 126
- Associating MIB Views with an SNMP User Group on page 151
- community on page 266

write-view

Supported Platforms ACX Series, EX Series, M Series, MX Series, OCX1100, PTX Series, QFabric System, QFX Series standalone switches, SRX Series, T Series

Syntax write-view *view-name*;

Hierarchy Level [edit snmp v3 vacm access group *group-name* (default-context-prefix | context-prefix *context-prefix*) security-model (any | usm | v1 | v2c) security-level (authentication | none | privacy)]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.0 for EX Series switches.
Statement introduced in Junos OS Release 11.1 for the QFX Series switches.
Command introduced in Junos OS Release 14.1X53-D20 for the OCX Series.

Description Associate the write view with a community (for SNMPv1 or SNMPv2c clients) or a group name (for SNMPv3 clients).

Options ***view-name***—Name of the view for which the SNMP user group has write permission.

Required Privilege Level snmp—To view this statement in the configuration.
snmp-control—To add this statement to the configuration.

Related Documentation
- Configuring MIB Views on page 126
- Configuring the Write View on page 152

CHAPTER 17

Operational Commands

- show snmp health-monitor
- show snmp health-monitor routing-engine history
- show snmp health-monitor routing-engine status
- show snmp mib (View)
- show system log-vital

show snmp health-monitor

Supported Platforms LN Series, SRX100, SRX110, SRX210, SRX220, SRX240, SRX550, SRX650, vSRX

Syntax
show snmp health-monitor
<alarms <detail>> | <logs>

Release Information Statement introduced in Junos OS Release 12.1X44-D10 for SRX Series devices.

Description Display information about SNMP health monitor alarms and logs.

Options
none—Display information about all health monitor alarms and logs.

alarms <detail>—(Optional) Display detailed information about health monitor alarms.

logs—(Optional) Display information about health monitor logs.

Required Privilege Level view

List of Sample Output
show snmp health-monitor on page 348
show snmp health-monitor alarms detail on page 349
show snmp health-monitor alarms brief on page 350

Output Fields Table 26 on page 346 describes the output fields for the **show snmp health-monitor** command. Output fields are listed in the approximate order in which they appear.

Table 26: show snmp health-monitor Output Fields

Field Name	Field Description
Alarm Index	Alarm identifier.
Variable description	Description of the health monitor object instance being monitored.
Variable name	Name of the health monitor object instance being monitored.
Value	Current value of the monitored variable in the most recent sample interval.

Table 26: show snmp health-monitor Output Fields *(continued)*

Field Name	Field Description

State	State of the alarm or event entry:
	• Alarms:
	  • **moderate-threshold**—Percentage of moderate threshold level resource utilization.
	  • **high-threshold** —Percentage of high-threshold level resource utilization.
	  • **critical-threshold** —Percentage of citical threshold level resource utilization.
	  • **active**—Entry is fully configured and activated.
	  • **falling threshold crossed**—Value of the variable has crossed the lower threshold limit.
	  • **rising threshold crossed**—Value of the variable has crossed the upper threshold limit.
	  • **under creation**—Entry is being configured and is not yet activated.
	  • **startup**—Alarm is waiting for the first sample of the monitored variable.
	  • **object not available**—Monitored variable of that type is not available to the health monitor agent.
	  • **instance not available**—Monitored variable's instance is not available to the health monitor agent.
	  • **object type invalid**—Monitored variable is not a numeric value.
	  • **object processing errored**—An error occurred when the monitored variable was processed.
	  • **unknown**—State is not one of the above.

Variable OID	Object ID to which the variable name is resolved. The format is **x.x.x.x**.
Sample type	Method of sampling the monitored variable and calculating the value to compare against the upper and lower thresholds. It can have the value of **absolute value** or **delta value**.

Startup alarm	Alarm that might be sent when this entry is first activated, depending on the following criteria:
	• Alarm is sent when one of the following situations exists:
	  • Value of the alarm is above or equal to the rising threshold and the startup type is either **rising alarm** or **rising or falling alarm.**
	  • Value of the alarm is below or equal to the falling threshold and the startup type is either **falling alarm** or **rising or falling alarm**.
	• Alarm is *not* sent when one of the following situations exists:
	  • Value of the alarm is above or equal to the rising threshold and the startup type is **falling alarm**.
	  • Value of the alarm is below or equal to the falling threshold and the startup type is **rising alarm**.
	  • Value of the alarm is between the thresholds.

Owner	Name of the entry configured by the user. If the entry was created through the CLI, the owner has **monitor** prepended to it.
Creator	Mechanism by which the entry was configured (**Health Monitor**).

Table 26: show snmp health-monitor Output Fields *(continued)*

Field Name	Field Description
Sample interval	Time period between samples (in seconds).
Rising threshold	Upper limit threshold value as a percentage of the maximum possible value.
Falling threshold	Lower limit threshold value as a percentage of the maximum possible value.
Rising event index	Event triggered when the rising threshold is crossed.
Falling event index	Event triggered when the falling threshold is crossed.

Sample Output

show snmp health-monitor

```
user@host> show snmp health-monitor

Alarm
Index  Variable description                              Value State

 32770 Health Monitor: md3:/jail/mfs utilization
       jnxHrStoragePercentUsed.16                          0 active

 32773 Health Monitor: md2:/mfs/var/run/utm utilization
       jnxHrStoragePercentUsed.15                          0 active

 32776 Health Monitor: md1:/mfs utilization
       jnxHrStoragePercentUsed.11                         11 active

 32779 Health Monitor: /var file system utilization
       jnxHrStoragePercentUsed.10                         44 critical threshold

 32782 Health Monitor: root file system utilization
       jnxHrStoragePercentUsed.1                          52 critical threshold

 32785 Health Monitor: /config file system utilization
       jnxHrStoragePercentUsed.2                           0 active

 32788 Health Monitor: RE 0 CPU utilization
       jnxOperatingCPU.9.1.0.0                            20 active

 32791 Health Monitor: RE 0 memory utilization
       jnxOperatingBuffer.9.1.0.0                         52 active

 32792 Health Monitor: Max Kernel Memory Used (%)
       jnxBoxKernelMemoryUsedPercent.0                     3 active

 32793 Health Monitor: jroute daemon memory usage
       Routing protocols process                       51452 active
       Management process                              38284 active
       Periodic packet management process               9828 active
       Bidirectional Forwarding Detection process      13088 active
       Service Deployment Client                       10012 active
       Event processing process                        12692 active
       Layer 2 address flooding and learning process      20212 active
```

```
    MPLS Periodic Traceroute process              10488 active
    Multicast Snooping process                     9608 active
    Feature license management process            12372 active
```

show snmp health-monitor alarms detail

```
user@host> show snmp health-monitor alarms detail

   Alarm Index 32770:
  Variable name                       jnxHrStoragePercentUsed.16
  Variable OID                        1.3.6.1.4.1.2636.3.31.1.1.1.1.16
  Sample type                         absolute value
  Startup alarm                       rising alarm
  Owner                               Health Monitor: md3:/jail/mfs utilization

  Creator                             Health Monitor
  State                               active
  Sample interval                  15 seconds
  Moderate threshold               20
  High threshold                   30
  Critical threshold               40
  Rising event index            32768
  Falling event index           32768
    Instance Value: 0
    Instance State: active

Alarm Index 32773:
  Variable name                       jnxHrStoragePercentUsed.15
  Variable OID                        1.3.6.1.4.1.2636.3.31.1.1.1.1.15
  Sample type                         absolute value
  Startup alarm                       rising alarm
  Owner                               Health Monitor: md2:/mfs/var/run/utm
                                      utilization
  Creator                             Health Monitor
  State                               active
  Sample interval                  15 seconds
  Moderate threshold               20
  High threshold                   30
  Critical threshold               40
  Rising event index            32768
  Falling event index           32768
    Instance Value: 0
    Instance State: active

Alarm Index 32793:
  Variable name                       sysApplElmtRunMemory.5
  Variable OID                        1.3.6.1.2.1.54.1.2.3.1.10.5
  Sample type                         absolute value
  Startup alarm                       rising alarm
  Owner                               Health Monitor: jroute daemon memory
                                      usage
  Creator                             Health Monitor
  State                               active
  Sample interval                  20 seconds
  Rising threshold              104857
  Falling threshold              91750
  Rising event index             32768
  Falling event index            32768
    Instance Name: sysApplElmtRunMemory.5.5.1258
    Instance Description: Routing protocols process
```

```
Instance Value: 51452
Instance State: active

Instance Name: sysApplElmtRunMemory.5.6.1255
Instance Description: Management process
Instance Value: 38284
Instance State: active

Instance Name: sysApplElmtRunMemory.5.6.3816
Instance Description: Management process
Instance Value: 38352
Instance State: active

Instance Name: sysApplElmtRunMemory.5.8.3815
Instance Description: Command-line interface
Instance Value: 49108
Instance State: active
```

show snmp health-monitor alarms brief

```
user@host> show snmp health-monitor alarms brief
 32791 Health Monitor: RE 0 memory utilization
       jnxOperatingBuffer.9.1.0.0                        52 active

 32792 Health Monitor: Max Kernel Memory Used (%)
       jnxBoxKernelMemoryUsedPercent.0                    3 active

 32793 Health Monitor: jroute daemon memory usage
       Routing protocols process                      51452 active
       Management process                             38284 active
       Management process                             38356 active
       Command-line interface                         49108 active
       Periodic packet management process              9828 active
       Bidirectional Forwarding Detection process     13088 active
       Service Deployment Client                      10012 active
       Event processing process                       12692 active
       Layer 2 address flooding and learning process      20212 active
       MPLS Periodic Traceroute process               10488 active
       Multicast Snooping process                      9608 active
       Feature license management process             12372 active

 32794 Health Monitor: jkernel daemon memory usage
       Init daemon                                     1684 active
       Chassis control process                       115888 rising threshold
       Firewall process                               22584 active
       Interface control process                      34000 active
       Simple Network Management Protocol process     21772 active
       Management Information Base II process         27848 active
       Alarm control process                          12568 active
       Packet Forwarding Engine statistics management process      24388 active
       Craft interface I/O control process            13248 active
       Remote operations process                      13712 active
       Class-of-service process                       18908 active
       Internal routing service process                7924 active
       Inet process                                    6052 active
       USB supervise process                           2388 active
       PPP process                                     8772 active
       Juniper Stateful Redundancy Protocol Daemon     13668 active
       Network security daemon                        24248 active
       Simple Mail Transfer Protocol Client process       8088 active
```

```
      PFE relay process                                8044 active
      Subscriber management process                   17852 active
      Subscriber management helper process            21076 active
      Web management gatekeeper process               12820 active
      Application-identification process              18328 active
      IDP policy daemon                               30188 active
      Shared memory routing socket message database process      15672 active
      System Health Management Daemon                 15004 active
      Network security trace daemon                   10400 active
      Wireless WAN process                            15016 active
      Wireless LAN service process                    13936 active

32797 Health Monitor: RE Temperature
      jnxFruTemp.9.1.0.0                                51 active

32800 Health Monitor: RE Process count usage
      hrSystemProcesses.0                              123 moderate threshold

32803 Health Monitor: RE Open file Descriptor count
      jnxHrSystemOpenFiles.0                           738 active

32804 Health Monitor: FWDD Micro-Kernel threads total CPU Utilization
      jnxFwddMicroKernelCPUUsage.0                      11 active

32805 Health Monitor: FWDD Real-Time threads total CPU Utilization
      jnxFwddRtThreadsCPUUsage.0                         0 active

32806 Health Monitor: FWDD DMA Memory utilization
      jnxFwddDmaMemUsage.0                               1 active

32807 Health Monitor: FWDD Heap utilization
      jnxFwddHeapUsage.0                                39 active
```

show snmp health-monitor routing-engine history

Supported Platforms LN Series, SRX100, SRX110, SRX210, SRX220, SRX240, SRX550, SRX650, vSRX

Syntax show snmp health-monitor routing-engine history resource <*cpu* | *memory* | *open-files-count* | *process-count* | *storage* | *temperature*>;

Release Information Statement introduced in Junos OS Release 12.1X44-D10 for branch SRX Series devices. Statement modified in Junos OS Release 12.1X45-D10.

Description Display the health-monitoring information collected for a Routing Engine.

Options

brief—Displays brief health monitor history.

extensive—Displays extensive health monitor history.

terse —Displays terse health monitor history.

Required Privilege Level view

Related Documentation

- show snmp health-monitor on page 346

List of Sample Output

show snmp health-monitor routing-engine history on page 353
show snmp health-monitor routing-engine history extensive on page 354
show snmp health-monitor routing-engine history terse on page 355

Output Fields Table 27 on page 352 describes the output fields for the **show snmp health-monitor routing engine history** command. Output fields are listed in the approximate order in which they appear.

Table 27: show snmp health-monitor routing engine history Output Fields

Field Name	Field Description
Resource	Name of the health monitor object instance being monitored.
Event	Displays the latest event and time associated with the resource. The available events are: • Moderate Rising • High Rising • Critical Rising • Moderate Falling • High Falling • Critical Falling

Table 27: show snmp health-monitor routing engine history Output Fields *(continued)*

Field Name	Field Description
Configuration	Effective configuration of a resource. • **interval** — Configured interval in seconds. • **moderate-threshold**—Percentage of moderate threshold level resource utilization. • **high-threshold** — Percentage of high-threshold level resource utilization. • **critical-threshold** — Percentage of critical threshold level resource utilization. • **action** — Configured action for a resource.
Usage Trail	Displays the previous usage records.
Top daemon	List of processes with high resource utilization.
Growing daemons	List of processes with high incremental resource utilization from the previous sample.
Top files	List of large files in a partition.
Growing files	List of files in a partition that have gotten larger since the previous sample.
Resource name	Name of the resource.
Latest event	Displays the latest event associated with the resource. The available events are: • Moderate Rising • High Rising • Critical Rising • Moderate Falling • High Falling • Critical Falling
Time elapsed	Displays the time elapsed since the event occurred.
Action	Displays the action associated with the resource. The available actions are: • Monitor • Prevent • Recover

Sample Output

show snmp health-monitor routing-engine history

```
user@host> show snmp health-monitor routing-engine history brief
Resource : CPU (jnxOperatingCPU.9.1.0.0)
    Event         : Critical Falling (76 %)         2013-04-10 18:44:47 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 76 76 76 78 78 78 78 78 78 78 ...
        Top and Growing Consumer (%)
            Top Consumer             Usage        Growth
            flowd_octeon_hm          252          2
```

```
                idle: cpu0                34          34
                av_worker                 3           2
                Growing Consumer          Usage       Growth
                idle: cpu0                34          34
                flowd_octeon_hm           252         2
                av_worker                 3           2
                Load averages:  2.01 (1 min)   1.70 (5 min)  2.01 (15 min)

Resource : Var:/cf/var (jnxHrStoragePercentUsed.5)
    Event         : High Rising (70 %)              2013-04-10 14:51:29 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 70 70 69 69 69 69 69 69 69 69 ...
        Top and Growing Consumer (KB)
            Top Consumer              Usage       Growth
            secdb_06.db               50424       0
            idpd_trace                23860       0
            SignatureUpdate.xml       20322       0
            ai_cachedfa_group_c       10784       0
            dfa_group_cache.db        10456       0
            Growing Consumer          Usage       Growth
            default-log-message       4403        4403
            chassisd                  1467        4
            jsrpd                     1202        2
            Storage used: 226034 KB, Inodes used: 506 Nodes
```

show snmp health-monitor routing-engine history extensive

```
user@host> show snmp health-monitor routing-engine history extensive
Resource : CPU (jnxOperatingCPU.9.1.0.0)
    Event         : Critical Falling (76 %)         2013-04-10 18:44:47 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 76 76 76 78 78 78 78 78 78 78 ...
        Top and Growing Consumer (%)
            Top Consumer              Usage       Growth
            flowd_octeon_hm           252         2
            idle: cpu0                34          34
            av_worker                 3           2
            Growing Consumer          Usage       Growth
            idle: cpu0                34          34
            flowd_octeon_hm           252         2
            av_worker                 3           2
            Load averages:  2.01 (1 min)   1.70 (5 min)  2.01 (15 min)

Resource : CPU (jnxOperatingCPU.9.1.0.0)
    Event         : Critical Rising (85 %)          2013-04-10 18:43:28 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 85 85 85 84 84 84 84 84 84 84 ...
        Top and Growing Consumer (%)
            Top Consumer              Usage       Growth
            flowd_octeon_hm           250         -1
            syshmd                    14          0
            cli                       8           0
            av_worker                 2           0
            av_worker                 1           0
            Load averages:  3.26 (1 min)   1.69 (5 min)  3.26 (15 min)

Resource : CPU (jnxOperatingCPU.9.1.0.0)
    Event         : High Rising (72 %)              2013-04-10 18:43:28 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 72 69 69 69 69 69 69 69 69 69 ...
        Top and Growing Consumer (%)
            Top Consumer              Usage       Growth
            flowd_octeon_hm           251         4
```

```
                init                    14          14
                syshmd                  14          14
                cli                     8           8
                av_worker               2           2
                Growing Consumer        Usage       Growth
                syshmd                  14          14
                init                    14          14
                cli                     8           8
                flowd_octeon_hm         251         4
                av_worker               2           2
                Load averages:  3.26 (1 min)   1.69 (5 min)  3.26 (15 min)

Resource : Var:/cf/var (jnxHrStoragePercentUsed.5)
    Event         : High Rising (70 %)              2013-04-10 14:51:29 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 70 70 69 69 69 69 69 69 69 69 ...
        Top and Growing Consumer (KB)
            Top Consumer            Usage       Growth
            secdb_06.db             50424       0
            idpd_trace              23860       0
            SignatureUpdate.xml     20322       0
            ai_cachedfa_group_c     10784       0
            dfa_group_cache.db      10456       0
            Growing Consumer        Usage       Growth
            default-log-message     4403        4403
            chassisd                1467        4
            jsrpd                   1202        2
            Storage used: 226034 KB, Inodes used: 506 Nodes

Resource : Var:/cf/var (jnxHrStoragePercentUsed.5)
    Event         : Moderate Rising (65 %)          2013-04-10 14:16:42 JST
    Configuration : 1/30/70/85/Monitor (Inter/Mod/High/Crit/Action)
        Usage Trail (%): 65 ...
        Top and Growing Consumer (KB)
            Top Consumer            Usage       Growth
            secdb_06.db             50424       0
            idpd_trace              23860       0
            SignatureUpdate.xml     20322       0
            ai_cachedfa_group_c     10784       0
            dfa_group_cache.db      10456       0
            Growing Consumer        Usage       Growth
            chassisd                1463        18
            jsrpd                   1200        7
            Storage used: 211868 KB, Inodes used: 503 Nodes
```

show snmp health-monitor routing-engine history terse

```
user@host> show snmp health-monitor routing-engine history terse

Resource name        Latest event     Time elapsed  Action

MD2:/mfs/var/run/utm High Falling     00:00:36      Monitor
Root:/cf             Moderate Rising  1d 02:25      Monitor
Var:/cf/var          Critical Rising  00:02:38      Monitor
CPU                  Critical Rising  1d 02:19      Monitor
Memory               Critical Rising  00:08:00      Monitor
RE process count     High Rising      1d 02:25      Monitor
RE open files count  Moderate Rising  1d 02:25      Monitor
RE Temperature       Moderate Rising  1d 02:24      Monitor
```

show snmp health-monitor routing-engine status

Supported Platforms SRX100, SRX110, SRX210, SRX220, SRX240, SRX550, SRX650, vSRX

Syntax show snmp health-monitor routing-engine status;

Release Information Statement introduced in Junos OS Release 12.1X45-D10 for branch SRX Series devices.

Description Display the SNMP health-monitoring information for a Routing Engine.

Required Privilege Level view

Related Documentation
- show snmp health-monitor routing-engine history on page 352

List of Sample Output show snmp health-monitor routing-engine status on page 356

Output Fields Table 28 on page 356 describes the output fields for the **show snmp health-monitor routing-engine status** command. Output fields are listed in the approximate order in which they appear.

Table 28: show snmp health-monitor routing engine status Output Fields

Field Name	Field Description
Alarm Index	Alarm identifier.
Resource name	Name of the resource.
Current State	Current state of the monitored variable.
Config Action	Displays the configured action.
Threshold	Displays the threshold value for medium, high, and critical as a percentage.
Interval	Displays the time taken in seconds.

Sample Output

show snmp health-monitor routing-engine status

```
user@host> show snmp health-monitor routing-engine status

Health monitor status

Alarm  Resource             Current        Config   Threshold Interval
Index  Name                 State          Action   (M/H/C)%  (sec)

32770  MD3:/jail/mfs        Active(47)     Monitor  70/80/90  1
32773  MD2:/mfs/var/run/utm Moderate(69)   Monitor  70/80/90  1
32776  MD1:/mfs             Active(13)     Monitor  70/80/90  1
32782  Root:/cf             Moderate(54)   Monitor  30/70/85  1
32785  Config:/config       Active(0)      Monitor  30/70/85  1
```

```
32779  Var:/cf/var          Critical(85)   Monitor  30/70/85  1
32788  CPU                  Critical(100)  Monitor  30/70/85  1
32791  Memory               Critical(88)   Monitor  70/80/90  1
32800  RE process count     High(81)       Monitor  30/70/85  1
32803  RE open files count  Moderate(58)   Monitor  30/70/85  1
32797  RE Temperature       Moderate(44)   Monitor  30/70/85  1
```

show snmp mib (View)

Supported Platforms LN Series, SRX Series

Syntax show snmp mib (get | get-next | walk) (ascii | decimal) *object-id*

Release Information Command introduced in Junos OS Release 9.4. Support for IPv4 and IPv6 systemwide policy statistics added in Junos OS Release 12.1X46-D10.

Description Display local SNMP MIB object values.

Options

get—Retrieve and display one or more SNMP object values.

get-next—Retrieve and display the next SNMP object values.

walk—Retrieve and display the SNMP object values that are associated with the requested object identifier (OID). When you use this option, the Junos OS displays the objects below the subtree that you specify.

ascii—Display the SNMP object's string indices as an ASCII-key representation.

decimal—Display the SNMP object values in the decimal (default) format. The **decimal** option is the default option for this command. Therefore, issuing the **show snmp mib (get | get-next | walk) decimal object-id** and the **show snmp mib (get | get-next | walk) object-id** commands display the same output.

object-id—The object can be represented by a sequence of dotted integers (such as **1.3.6.1.2.1.2**) or by its subtree name (such as **interfaces**). When entering multiple objects, enclose the objects in quotation marks.

NOTE: On all high-end SRX Series devices, the **show snmp mib** command will not display the output for security related MIBs. We recommend that you use an SNMP client and prefix **logical-system-name@** to the community name. For example, if the community is public, use **default@public** for default root logical system.

Required Privilege Level snmp—To view this statement in the configuration.

Related Documentation

- *SNMP MIBs and Traps Monitoring and Troubleshooting Guide for Security Devices*

List of Sample Output

Output Fields Table 29 on page 359 describes the output fields for the **show snmp mib** command. Output fields are listed in the approximate order in which they appear.

Table 29: show snmp mib Output Fields

Field Name	Field Description
name	Object name and numeric instance value.
object value	Object value. The Junos OS translates OIDs into the corresponding object names.

Sample Output

show snmp mib walk (standalone)

```
user@host> show snmp mib walk jnxJsSPUMonitoringObjectsTable
jnxJsSPUMonitoringFPCIndex.5 = 5
jnxJsSPUMonitoringSPUIndex.5 = 0
jnxJsSPUMonitoringCPUUsage.5 = 0
jnxJsSPUMonitoringMemoryUsage.5 = 61
jnxJsSPUMonitoringCurrentFlowSession.5 = 0
jnxJsSPUMonitoringMaxFlowSession.5 = 524288
jnxJsSPUMonitoringCurrentCPSession.5 = 0
jnxJsSPUMonitoringMaxCPSession.5 = 2359296
jnxJsSPUMonitoringNodeIndex.5 = 0
jnxJsSPUMonitoringNodeDescr.5 = single
```

show snmp mib walk (HA)

```
user@switch> show snmp mib walk jnxJsSPUMonitoringObjectsTable
jnxJsSPUMonitoringFPCIndex.20 = 5
jnxJsSPUMonitoringFPCIndex.21 = 5
jnxJsSPUMonitoringFPCIndex.44 = 5
jnxJsSPUMonitoringFPCIndex.45 = 5
jnxJsSPUMonitoringSPUIndex.20 = 0
jnxJsSPUMonitoringSPUIndex.21 = 1
jnxJsSPUMonitoringSPUIndex.44 = 0
jnxJsSPUMonitoringSPUIndex.45 = 1
jnxJsSPUMonitoringCPUUsage.20 = 0
jnxJsSPUMonitoringCPUUsage.21 = 0
jnxJsSPUMonitoringCPUUsage.44 = 0
jnxJsSPUMonitoringCPUUsage.45 = 0
jnxJsSPUMonitoringMemoryUsage.20 = 64
jnxJsSPUMonitoringMemoryUsage.21 = 60
jnxJsSPUMonitoringMemoryUsage.44 = 64
jnxJsSPUMonitoringMemoryUsage.45 = 60
jnxJsSPUMonitoringCurrentFlowSession.20 = 0
jnxJsSPUMonitoringCurrentFlowSession.21 = 1
jnxJsSPUMonitoringCurrentFlowSession.44 = 0
jnxJsSPUMonitoringCurrentFlowSession.45 = 1
jnxJsSPUMonitoringMaxFlowSession.20 = 421888
jnxJsSPUMonitoringMaxFlowSession.21 = 843776
jnxJsSPUMonitoringMaxFlowSession.44 = 421888
jnxJsSPUMonitoringMaxFlowSession.45 = 843776
jnxJsSPUMonitoringCurrentCPSession.20 = 1
jnxJsSPUMonitoringCurrentCPSession.21 = 0
jnxJsSPUMonitoringCurrentCPSession.44 = 1
jnxJsSPUMonitoringCurrentCPSession.45 = 0
```

```
jnxJsSPUMonitoringMaxCPSession.20 = 2359296
jnxJsSPUMonitoringMaxCPSession.21 = 0
jnxJsSPUMonitoringMaxCPSession.44 = 2359296
jnxJsSPUMonitoringMaxCPSession.45 = 0
jnxJsSPUMonitoringNodeIndex.20 = 0
jnxJsSPUMonitoringNodeIndex.21 = 0
jnxJsSPUMonitoringNodeIndex.44 = 1
jnxJsSPUMonitoringNodeIndex.45 = 1
jnxJsSPUMonitoringNodeDescr.20 = node0
jnxJsSPUMonitoringNodeDescr.21 = node0
jnxJsSPUMonitoringNodeDescr.44 = node1
jnxJsSPUMonitoringNodeDescr.45 = node1
```

show snmp mib walk jnxJsPolicySystemStats

```
user@host> show snmp mib walk jnxJsPolicySystemStats
jnxJsPolicySystemStatsTotalAllowIPv4Packets.0 = 10347
jnxJsPolicySystemStatsTotalAllowIPv4Bytes.0 = 94053327
jnxJsPolicySystemStatsTotalAllowIPv4PacketsRate.0 = 21
jnxJsPolicySystemStatsTotalAllowIPv4BytesRate.0 = 1012
jnxJsPolicySystemStatsTotalDropIPv4Packets.0 = 257
jnxJsPolicySystemStatsTotalDropIPv4Bytes.0 = 40298
jnxJsPolicySystemStatsTotalDropIPv4PacketsRate.0 = 0
jnxJsPolicySystemStatsTotalDropIPv4BytesRate.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv4Flows.0 = 1
jnxJsPolicySystemStatsTotalAllowIPv4FlowsRate.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv6Packets.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv6Bytes.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv6PacketsRate.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv6BytesRate.0 = 0
jnxJsPolicySystemStatsTotalDropIPv6Packets.0 = 0
jnxJsPolicySystemStatsTotalDropIPv6Bytes.0 = 0
jnxJsPolicySystemStatsTotalDropIPv6PacketsRate.0 = 0
jnxJsPolicySystemStatsTotalDropIPv6BytesRate.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv6Flows.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv6FlowsRate.0 = 0
jnxJsPolicySystemStatsEnabled.0 = 1
```

show snmp mib walk jnxJsPolicySystemStatsIPv4

```
user@host> show snmp mib walk jnxJsPolicySystemStatsIPv4
jnxJsPolicySystemStatsTotalAllowIPv4Packets.0 = 10347
jnxJsPolicySystemStatsTotalAllowIPv4Bytes.0 = 94053327
jnxJsPolicySystemStatsTotalAllowIPv4PacketsRate.0 = 21
jnxJsPolicySystemStatsTotalAllowIPv4BytesRate.0 = 1012
jnxJsPolicySystemStatsTotalDropIPv4Packets.0 = 257
jnxJsPolicySystemStatsTotalDropIPv4Bytes.0 = 40298
jnxJsPolicySystemStatsTotalDropIPv4PacketsRate.0 = 0
jnxJsPolicySystemStatsTotalDropIPv4BytesRate.0 = 0
jnxJsPolicySystemStatsTotalAllowIPv4Flows.0 = 1
jnxJsPolicySystemStatsTotalAllowIPv4FlowsRate.0 = 0
```

show snmp mib walk jnxJsPolicySystemStatsTotalAllowIPv4Packets

```
user@host> show snmp mib walk jnxJsPolicySystemStatsTotalAllowIPv4Packets
jnxJsPolicySystemStatsTotalAllowIPv4Packets.0 = 10347
```

show system log-vital

Supported Platforms SRX Series

Syntax
show system log-vital
<data | oid | status>

Release Information Command introduced in Junos OS Release 12.1X47-D15.

Description Display the vital data of MIB OIDs.

Options
data—Display detailed vital data of the current day.

oid—Display configured OID or group.

status—Display the settings of the vital log.

Required Privilege Level view

Related Documentation
- log-vital on page 286

List of Sample Output
show system log-vital data on page 362
show system log-vital oid on page 363
show system log-vital status on page 363

Output Fields Table 30 on page 361 lists the output fields for the show system log-vital command. Output fields are listed in the approximate order in which they appear.

Table 30: show system log-vital Output fields

Field Name	Field Description
Node	Identification number of the node. It can be 0 or 1.
SPU	Identification of Services Processing Unit.
CPU	CPU usage of SPU in percentage.
Mem	Memory usage of SPU in percentage.
Flow-Sess	Number of flow sessions.
CP-Sess	Number of central point sessions.
IPv4-Sess	Number of IPv4 sessions.
IPv6-Sess	Number of IPv6 sessions.
CP-IPv4	Number of central point IPv4 sessions.

Table 30: show system log-vital Output fields *(continued)*

Field Name	Field Description
CP-IPv6	Number of central point IPv6 sessions.
OID list	OIDs that are being monitored.
OID number	Number of OIDs that are being monitored.
Group SPU list	SPUs that are being monitored.
Group SPU number	Number of SPUs that are being monitored.
Group screen list	Security zones whose screen stats are being monitored.
Group screen number	Number of security zones whose screen stats are being monitored.
Group	A set of OIDs. Once a group is enabled, all OIDs in the group are monitored.
interval	Number of minutes used for the data collection interval.
file-days	Number of days for the dump file to be stored.
storage-limit	Storage usage limit in percentage.
file-size	Size of the current dump file.
state	Number that indicates which state the current collection is in. It could indicate **IDLE** or **ONGOING**.
snmp mgmt-sock op number	Stat number of the querying MIB.
current timer counter	Number that indicates the collection timer.

Sample Output

show system log-vital data

```
user@host> show system log-vital data

#
# Start firefly-perimeter--"fw1" Vitals Check    Fri Sep  5 00:00:44 2014
#

[Fri Sep  5 00:00:44 2014] Vital data of SPU
Node    SPU          CPU   Mem   Flow-Sess   CP-Sess     IPv4-Sess   IPv6-Sess
 CP-IPv4      CP-IPv6
================================================================================
node0   fwdd         0     55    10          0           10          0
 0           0

#
# End firefly-perimeter--"fw1" Vitals Check    Fri Sep  5 00:00:45 2014
```

```
#

#
# Start firefly-perimeter--"fw1" Vitals Check    Fri Sep  5 00:01:45 2014
#

[Fri Sep  5 00:01:45 2014] Vital data of SPU
Node    SPU          CPU   Mem   Flow-Sess   CP-Sess     IPv4-Sess   IPv6-Sess
 CP-IPv4      CP-IPv6
================================================================================
node0   fwdd         0     55    16          0           16          0
 0            0

#
# End firefly-perimeter--"fw1" Vitals Check    Fri Sep  5 00:01:45 2014
#
```

show system log-vital oid

```
user@host> show system log-vital oid

OID list:
lldpLocSysName.0          sys-name
                  jnxJsNodeCurrentTotalSessIPv4.0          IPv4-sess-number
                 .1.3.6.1.4.1.2636.3.1.13.1.8.9.1.0.0          re cpu usage
OID number: 3
Group SPU list:
                 All
Group SPU number: 1
Group screen list:
                trust
                untrust
Group screen number: 2
Group:     idp cluster-counter storage operating
```

show system log-vital status

```
user@host> show system log-vital status

log vital status:
interval: 1 Minutes
file-days: 4 days
storage-limit: 75 percent
file-size: 3 Mbytes
state: 5
snmp mgmt-sock op number: 0
current timer counter: 1 (vs 60)
```

PART 6

Index

Index

Symbols

A

B

C

M

N

O

P

R

S

T

U

V

W