JUNIPER NETWORKS | Engineering Simplicity

Junos® OS

Network Management and Monitoring Guide

Published
2023-12-15

JUNOS

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

Juniper Networks, the Juniper Networks logo, Juniper, and Junos are registered trademarks of Juniper Networks, Inc. in the United States and other countries. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners.

Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

Junos® OS Network Management and Monitoring Guide
Copyright © 2023 Juniper Networks, Inc. All rights reserved.

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 https://support.juniper.net/support/eula/. By downloading, installing or using such software, you agree to the terms and conditions of that EULA.

Table of Contents

About This Guide | xxxvi

1 Overview

Device Management Functions in Junos OS | 2

Device and Network Management Features | 5

Tracing and Logging Operations | 10

Junos Space Support for Network Management | 12

Diagnostic Tools Overview | 13

2 Operation, Administration, and Management Features

Ethernet OAM and Connectivity Fault Management for Routers | 19

Introduction to OAM Connectivity Fault Management (CFM) | 19

- Ethernet OAM Connectivity Fault Management | 20
- IEEE 802.1ag OAM Connectivity Fault Management | 21

Configure Connectivity Fault Management (CFM) | 26

- Create a Maintenance Domain | 27
- Create a Maintenance Association | 28
- Configure Maintenance Intermediate Points (MIPs) | 29
- Configure Maintenance Association Intermediate Points in ACX Series | 31
- Configure a MEP to Generate and Respond to CFM Protocol Messages | 35
  - Configure a Maintenance Association End Point (MEP) | 36
  - Configure a Remote Maintenance Association End Point (MEP) | 38
- Configure Service Protection for VPWS over MPLS Using the MEP Interface | 39
- Configure Linktrace Protocol in CFM | 44
- Continuity Check Protocol Parameters Overview | 45
- Configuring Continuity Check Protocol Parameters for Fault Detection | 46
- Configuring Rate Limiting of Ethernet OAM Messages | 47
- Enabling Enhanced Connectivity Fault Management Mode | 50
- Configure Connectivity Fault Management for Interoperability During Unified In-Service Software Upgrades | 52

Junos OS Support for Performance Monitoring Compliant with Technical Specification MEF 36 | 53

Damping CFM performance Monitoring Traps and Notifications to Prevent Congestion of The NMS | 54

CFM Action Profile | 55

CFM Action Profile to Bring Down a Group of Logical Interfaces Overview | 55

Configure a CFM Action Profile to Bring Down a Group of Logical Interfaces | 57

Configure a CFM Action Profile to Specify CFM Actions for CFM Events | 61

Ethernet Local Management Interface | 62

Ethernet Local Management Interface Overview | 62

Configure the Ethernet Local Management Interface | 65

Example E-LMI Configuration | 67

CFM Support for CCC Encapsulated Packets | 73

IEEE 802.1ag CFM OAM Support for CCC Encapsulated Packets Overview | 73

CFM Features Supported on Layer 2 VPN Circuits | 73

Configure CFM for CCC Encapsulated Packets | 74

Configure Unified ISSU for 802.1ag CFM | 75

CFM Monitoring between CE and PE Devices | 79

CFM Action Profile Asynchronous Notification | 80

Configuring a CFM Action Profile to Asyncronus Notification | 80

Understanding CFM Monitoring between CE and PE Devices | 83

Configuring Port Status TLV and Interface Status TLV | 85

TLVs Overview | 85

Various TLVs for CFM PDUs | 86

Support for Additional Optional TLVs | 88

MAC Status Defects | 96

Configuring Remote MEP Action Profile Support | 98

Monitoring a Remote MEP Action Profile | 99

Configuring Chassis ID TLV | 100

Configuring MAC Flush Message Processing in CET Mode | 101

Example: Configuring an Action Profile Based on Connection Protection TLVs | 104

Requirements | 104

Overview and Topology | 104

Configuration | 105

Configure Continuity Check Messages | **107**

Configure Faster Protection Switching for Point-to-Point Network Topologies | **108**

Configure Faster Convergence for Dual-Homed Multipoint-to-Multipoint Network Topologies | **110**

Configure a Primary VLAN ID for Increased Flexibility | **111**

Configure a Remote Maintenance Association to Accept a Different ID | **112**

Example: Configure Ethernet CFM on Physical Interfaces | **114**

Requirements | **114**

Overview | **114**

Configuration | **114**

Example: Configure Ethernet CFM on Bridge Connections | **117**

Example: Configure Ethernet CFM over VPLS | **122**

Link Fault Management for Routers | 132

Introduction to OAM Link Fault Management (LFM) | **132**

IEEE 802.3ah OAM Link-Fault Management Overview | **132**

Understanding Ethernet OAM Link Fault Management for ACX Series Routers | **133**

Configuring Ethernet 802.3ah OAM | **135**

Configure Link Fault Management | **137**

Configuring Link Discovery | **138**

Configuring the OAM PDU Interval | **138**

Configuring the OAM PDU Threshold | **139**

Configuring Threshold Values for Local Fault Events on an Interface | **139**

Disabling the Sending of Link Event TLVs | **140**

Example: Configuring IEEE 802.3ah OAM Support on an Interface | **140**

Example: Configuring IEEE 802.3ah OAM Support for an Interface on ACX Series | **141**

Requirements | **141**

Overview and Topology | **142**

Configuring IEEE 802.3ah OAM on an ACX Series Router | **142**

Example: Configuring Ethernet LFM Between Provider Edge and Customer Edge | **145**

Example: Configuring Ethernet LFM for CCC | **146**

Example: Configuring Ethernet LFM for Aggregated Ethernet | **148**

Configuring an OAM Action Profile | **151**

Specifying the Actions to Be Taken for Link-Fault Management Events | **152**

Monitoring the Loss of Link Adjacency | **153**

Monitoring Protocol Status | 153

Configuring Threshold Values for Fault Events in an Action Profile | 154

Applying an Action Profile | 154

Remote Fault Detection for Link Fault Management | 155

Detecting Remote Faults | 155

Enabling Dying Gasp Functionality | 156

Remote Loopback for Link Fault Management | 158

Setting a Remote Interface into Loopback Mode | 158

Enabling Remote Loopback Support on the Local Interface | 159

Enabling Nonstop Routing for Ethernet Link Fault Management on Backup Routers | 159

Example: Configuring Ethernet LFM with Loopback Support | 163

Ethernet OAM Link Fault Management for Switches | 167

Ethernet OAM Link Fault Management | 167

Configure Ethernet OAM Link Fault Management | 168

Example: Configure Ethernet OAM Link Fault Management | 172

Requirements | 172

Overview and Topology | 173

Configuring Ethernet OAM Link Fault Management on Switch 1 | 173

Configuring Ethernet OAM Link Fault Management on Switch 2 | 175

Verification | 177

Ethernet OAM Connectivity Fault Management for Switches | 179

Understand Ethernet OAM Connectivity Fault Management for Switches | 179

Configure Ethernet OAM Connectivity Fault Management (CLI Procedure) | 182

Creating the Maintenance Domain | 183

Configuring the Maintenance Domain MIP Half Function | 184

Creating a Maintenance Association | 184

Configuring the Continuity Check Protocol | 184

Configuring a Maintenance Association End Point | 185

Configuring a Connectivity Fault Management Action Profile | 187

Configuring the Linktrace Protocol | 187

Example: Configure Ethernet OAM Connectivity Fault Management on EX Series Switches | 188

Requirements | 189

Overview and Topology | **189**

Configuring Ethernet OAM Connectivity Fault Management on Switch 1 | **189**

Configuring Ethernet OAM Connectivity Fault Management on Switch 2 | **191**

Verification | **194**

Ethernet Frame Delay | 196

Ethernet Frame Delay Measurements on Switches | **196**

Configure MEP Interfaces on Switches to Support Ethernet Frame Delay Measurements (CLI Procedure) | **198**

Configure One-Way Ethernet Frame Delay Measurements on Switches (CLI Procedure) | **199**

Configure an Iterator Profile on a Switch (CLI Procedure) | **200**

Trigger an Ethernet Frame Delay Measurement Session on a Switch | **201**

Configure Two-Way Ethernet Frame Delay Measurements on Switches (CLI Procedure) | **202**

Ethernet Service OAM (ITU-TY.1731) for Routers | 204

ITU-T Y.1731 Ethernet Service OAM Overview | **204**

Ethernet Frame Delay Measurements Overview | **205**

Ethernet Frame Loss Measurement Overview | **212**

Service-Level Agreement Measurement | **213**

On-Demand Mode for SLA Measurement | **213**

Proactive Mode for SLA Measurement | **214**

Ethernet Failure Notification Protocol Overview | **216**

Ethernet Synthetic Loss Measurement Overview | **217**

Scenarios for Configuration of ETH-SLM | **218**

Format of ETH-SLM Messages | **219**

Transmission of ETH-SLM Messages | **221**

Configure Ethernet Frame Delay Measurement Sessions | **224**

Guidelines for Configuring Routers to Support an ETH-DM Session | **224**

Guidelines for Starting an ETH-DM Session | **226**

Guidelines for Managing ETH-DM Statistics and ETH-DM Frame Counts | **228**

Configuring Routers to Support an ETH-DM Session | **234**

Configuring MEP Interfaces | **234**

Ensuring That Distributed ppm Is Not Disabled | **235**

Enabling the Hardware-Assisted Timestamping Option | **238**

Configuring the Server-Side Processing Option | 238

Triggering an Ethernet Frame Delay Measurements Session | 239

Starting an ETH-DM Session | 241

Using the monitor ethernet delay-measurement Command | 241

Starting a One-Way ETH-DM Session | 242

Starting a Two-Way ETH-DM Session | 243

Example: Configuring One-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 244

Example: Configuring Two-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 250

Managing Continuity Measurement Statistics | 256

Displaying Continuity Measurement Statistics | 256

Clearing Continuity Measurement Statistics | 257

Viewing Ethernet Frame Delay Measurements Statistics | 258

Managing ETH-DM Statistics and ETH-DM Frame Counts | 259

Displaying ETH-DM Statistics Only | 260

Displaying ETH-DM Statistics and Frame Counts | 261

Displaying ETH-DM Frame Counts for MEPs by Enclosing CFM Entity | 262

Displaying ETH-DM Frame Counts for MEPs by Interface or Domain Level | 263

Clearing ETH-DM Statistics and Frame Counts | 264

Configuring MEP Interfaces to Support Ethernet Frame Delay Measurements | 266

Configure Ethernet Frame Loss Measurement | 268

Configuring Statistical Frame Loss Measurement for VPLS Connections | 268

Managing ETH-LM Statistics | 269

Displaying ETH-LM Statistics | 269

Clearing ETH-LM Statistics | 271

Example: Measuring Ethernet Frame Loss for Single-Tagged LMM/LMR PDUs | 272

Requirements | 272

Overview and Topology | 272

Configuration | 273

Verification | 286

Example: Measuring Ethernet Frame Loss for Dual-Tagged LMM/LMR PDUs | 288

Requirements | 289

Overview and Topology | 289

Configuration | 290

Verification | 303

Configure an Iterator Profile | 305

Configuring an Iterator Profile | 306

Verifying the Configuration of an Iterator Profile | 309

Displaying the Configuration of an Iterator Profile for Two-way Delay Measurement | 310

Displaying the Configuration of an Iterator Profile for Loss Measurement | 311

Displaying the Configuration of a Remote MEP with an Iterator Profile | 312

Disabling an Iterator Profile | 313

Managing Iterator Statistics | 314

Displaying Iterator Statistics | 314

Clearing Iterator Statistics | 321

Configuring a Remote MEP with an Iterator Profile | 322

Configure Ethernet Synthetic Loss Measurements | 324

Guidelines for Configuring ETH-SLM | 324

Starting a Proactive ETH-SLM Session | 326

Configuring MEP Interfaces | 326

Configuring an Iterator Profile for ETH-SLM | 327

Associating the Iterator Profile with MEPs for ETH-SLM | 329

Starting an On-Demand ETH-SLM Session | 331

Managing ETH-SLM Statistics and ETH-SLM Frame Counts | 332

Displaying ETH-SLM Statistics Only | 332

Displaying ETH-SLM Statistics and Frame Counts | 333

Displaying ETH-SLM Frame Counts for MEPs by Enclosing CFM Entity | 335

Displaying ETH-SLM Frame Counts for MEPs by Interface or Domain Level | 336

Clearing ETH-SLM Statistics and Frame Counts | 337

Clearing Iterator Statistics | 338

Troubleshooting Failures with ETH-SLM | 339

Ethernet Alarm Indication | 341

Ethernet Alarm Indication Signal (ETH-AIS) Function Overview | 341

Ethernet Alarm Indication Signal Overview | 346

Configuring ETH-AIS on a CFM MEP | 347

Configuring an Action Profile | 348

Configuring an Action to Be Taken When an AIS Alarm Is Detected | 350

Attaching the Action Profile to a CFM MEP | 351

Configuring Alarm Indication Signal on ACX Series Routers | 353

Inline Transmission Mode | 356

Enabling Inline Transmission of Continuity Check Messages for Maximum Scaling | 356

Enabling Inline Transmission of Link Fault Management Keepalives for Maximum Scaling | 357

Enabling Inline Mode Of Performance Monitoring To Achieve Maximum Scaling | 361

Supported Inline CCM and Inline PM Scaling Values | 364

3 Network Monitoring by using SNMP

SNMP Architecture and SNMP MIBs Overview | 369

Understand SNMP Implementation in Junos OS | 371

Loading MIB Files to a Network Management System | 375

Understand the Integrated Local Management Interface | 378

Configure SNMP in Junos OS | 379

Configure SNMP | 379

Configure SNMP Details | 388

Configure the Commit Delay Timer | 391

Configure SNMP on a Device Running Junos OS | 392

Example: Configure SNMP on the QFabric System | 394

Requirements | 394

Overview | 395

Configuration | 395

Configure Options on Managed Devices for Better SNMP Response Time | 399

Enable the stats-cache-lifetime Option | 399

Filter Out Duplicate SNMP Requests | 400

Exclude Interfaces That Are Slow in Responding to SNMP Queries | 400

Enterprise Specific Utility MIB to Enhance SNMP Coverage | 402

Utility MIB | 402

Optimize the Network Management System Configuration for the Best Results | 405

Interfaces to Accept SNMP Requests | 407

Configure the Interfaces on Which SNMP Requests Can Be Accepted | **407**

Configure a Proxy SNMP Agent | **407**

Example: Configure Secured Access List Checking | **408**

Filter Interface Information Out of SNMP Get and GetNext Output | **409**

Configure SNMP for Routing Instances | 410

Understand SNMP Support for Routing Instances | **411**

SNMPv3 Management Routing Instance | **412**

SNMP MIBs Supported for Routing Instances | **413**

Support Classes for MIB Objects | **425**

SNMP Traps Supported for Routing Instances | **426**

Identify a Routing Instance | **427**

Enable SNMP Access over Routing Instances | **428**

Specify a Routing Instance in an SNMPv1 or SNMPv2c Community | **428**

Example: Configure Interface Settings for a Routing Instance | **429**

Configure Access Lists for SNMP Access over Routing Instances | **431**

Configure SNMP Remote Operations | 432

SNMP Remote Operations Overview | **432**

Use the Ping MIB for Remote Monitoring Devices Running Junos OS | **436**

Start a Ping Test | **437**

- Before You Begin | **437**
- Start a Ping Test | **437**
- Use Multiple Set PDUs | **438**
- Use a Single Set PDU | **438**

Monitor a Running Ping Test | **438**

- pingResultsTable | **439**
- pingProbeHistoryTable | **440**
- Generate Traps | **441**

Gather Ping Test Results | **442**

Stop a Ping Test | **444**

Interpret Ping Variables | **444**

Use the Traceroute MIB for Remote Monitoring Devices Running Junos OS | **445**

Start a Traceroute Test | **445**

- Use Multiple Set PDUs | **446**
- Use a Single Set PDU | **446**

Monitor a Running Traceroute Test | **446**

- traceRouteResultsTable | **447**
- traceRouteProbeResultsTable | **448**
- traceRouteHopsTable | **449**
- Generate Traps | **450**

Monitor Traceroute Test Completion | **450**

Gather Traceroute Test Results | **451**

Stop a Traceroute Test | **453**

Interpret Traceroute Variables | **453**

SNMP Traps | 454

Configure SNMP Traps | **454**

Configure SNMP Trap Options | **456**

- Configure the Source Address for SNMP Traps | **457**
- Configure the Agent Address for SNMP Traps | **460**
- Add snmpTrapEnterprise Object Identifier to Standard SNMP Traps | **460**

Configure SNMP Trap Groups | **461**

Configure SNMP Trap Options and Groups on a Device Running Junos OS | **463**

Example: Configure SNMP Trap Groups | **464**

Manage Traps | **465**

SNMP Traps Supported by Junos OS | 467

SNMP Traps Support | **468**

Standard SNMP Traps Supported by Junos OS | **487**

Customized SNMP MIBs for Syslog Traps | **498**

- Overview of Custom SNMP MIBs | **498**
- Define a Custom MIB for a Syslog Trap | **500**
- Limitations of Using Custom SNMP Traps | **507**
- Example Custom Syslog Trap | **507**

Trace SNMP Activity | 514

Monitor SNMP Activity and Track Problems That Affect SNMP Performance on a Device Running Junos OS | **514**

- Check for MIB Objects Registered with SNMPd | **515**
- Track SNMP Activity | **516**
- Monitor SNMP Statistics | **516**
- Check CPU Utilization | **517**
- Check Kernel and Packet Forwarding Engine Response | **517**

Trace SNMP Activity on a Device Running Junos OS | **518**

- Configure the Number and Size of SNMP Log Files | **519**
- Configure Access to the Log File | **519**
- Configure a Regular Expression for Lines to Be Logged | **520**
- Configure the Trace Operations | **520**

Example: Tracing SNMP Activity | **522**

Configure the Certificate Expiration Trap | **523**

Enable Peer Down and IPsec Tunnel Down Traps | **524**

Access Privileges for an SNMP Group | 524

Configure the Access Privileges Granted to a Group | **525**

- Configure the Group | **526**
- Configure the Security Model | **526**
- Configure the Security Level | **527**
- Associate MIB Views with an SNMP User Group | **527**
  - Configure the Notify View | **528**
  - Configure the Read View | **528**
  - Configure the Write View | **528**

Example: Configure the Access Privileges Granted to a Group | **529**

Assign Security Model and Security Name to a Group | **530**

Configure the Security Model | 530

Assign Security Names to Groups | 531

Configure the Group | 531

Example: Security Group Configuration | 532

Configure Local Engine ID on SNMPv3 | 532

Configure SNMPv3 | 533

Create SNMPv3 Users | 535

Minimum SNMPv3 Configuration on a Device Running Junos OS | 535

Example: SNMPv3 Configuration | 536

Configure SNMPv3 Authentication Type and Encryption Type | 540

Configure SNMPv3 Authentication Type | 540

Configur MD5 Authentication | 541

Configure SHA Authentication | 541

Configure No Authentication | 541

Configure SNMPv3 Encryption Type | 542

Configure Advanced Encryption Standard Algorithm | 542

Configure Data Encryption Algorithm | 542

Configure Triple DES | 542

Configure No Encryption | 542

SNMPv3 Traps | 543

Configure SNMPv3 Traps on a Device Running Junos OS | 543

Configure SNMPv3 Trap Notification | 544

Example: Configure SNMPv3 Trap Notification | 544

Configure the Trap Notification Filter | 545

Configure the Trap Target Address | 545

Configure the Address | 546

Configure the Address Mask | 546

Configure the Port | 546

Configure the Routing Instance | 547

Configure the Trap Target Address | 547

Apply Target Parameters | 547

Example: Configure the Tag List | 547

Define and Configure the Trap Target Parameters | 548

Apply the Trap Notification Filter | 549

Configure the Target Parameters | 549

Configure the Message Processing Model | 549

Configure the Security Model | 549

Configure the Security Level | 550

Configure the Security Name | 550

SNMPv3 Informs | 550

Example: Configure the Inform Notification Type and Target Address | 552

Example: Configure the Remote Engine ID and Remote User | 553

Requirements | 553

Overview | 553

Configuration | 555

Verification | 556

SNMP Communities | 558

Configure SNMP Communities | 558

Add a Group of Clients to an SNMP Community | 562

Configure SNMP Community String | 564

Examples: Configure the SNMP Community String | 564

Configure the SNMPv3 Community | 566

Configuring the Community Name | 567

Configuring the Context | 567

Configuring the Security Names | 568

Configuring the Tag | 568

Example: Configure SNMPv3 Community | 568

Requirements | 568

Overview | 569

Configuration | 569

Verification | 572

MIB Views | 572

Configure MIB Views | **573**

Configure Ping Proxy MIB | **574**

SNMP MIBs Supported by Junos OS and Junos OS Evolved | 575

SNMP MIBs Support | **575**

MIB Objects for the QFX Series | **584**

Fabric Chassis MIB | **588**

Standard SNMP MIBs Supported by Junos OS | **594**

Enterprise-Specific MIBs for Junos OS Evolved | **609**

Enterprise-Specific SNMP MIBs Supported by Junos OS | **622**

Standard MIBs for Junos OS Evolved | **643**

Junos OS SNMP FAQs | 656

4 **Remote Network Monitoring (RMON) with SNMP Alarms and Events**

Remote Network Monitoring (RMON) | 687

RMON Overview | **687**

RMON Alarms and Events Configuration | **692**

Configure RMON Alarms and Events | **692**

- Configure SNMP | **693**
- Configure an Event | **694**
- Configure an Alarm | **695**

Monitor RMON MIB Tables | **696**

RMON MIB Event, Alarm, Log, and History Control Tables | **697**

Minimum RMON Alarm and Event Entry Configuration | **700**

Configure an RMON Alarm Entry and Its Attributes | **701**

- Configure the Alarm Entry | **701**
- Configure the Description | **702**
- Configure the Falling Event Index or Rising Event Index | **702**
- Configure the Falling Threshold or Rising Threshold | **702**

Configure the Interval | 703
Configure the Falling Threshold Interval | 703
Configure the Request Type | 704
Configure the Sample Type | 704
Configure the Startup Alarm | 705
Configure the System Log Tag | 705
Configure the Variable | 705

Configure an RMON Event Entry and Its Attributes | 706

Example: Configure an RMON Alarm and Event Entry | 707

Use alarmTable to Monitor MIB Objects | 707

Create an Alarm Entry | 708
Configure the Alarm MIB Objects | 708
alarmInterval | 709
alarmVariable | 709
alarmSampleType | 709
alarmValue | 709
alarmStartupAlarm | 710
alarmRisingThreshold | 710
alarmFallingThreshold | 710
alarmOwner | 711
alarmRisingEventIndex | 711
alarmFallingEventIndex | 711
Activate a New Row in alarmTable | 711
Modify an Active Row in alarmTable | 711
Deactivate a Row in alarmTable | 712

Use eventTable to Log Alarms | 712

Create an Event Entry | 712
Configure the MIB Objects | 713
eventType | 713
eventCommunity | 713
eventOwner | 714
eventDescription | 714
Activate a New Row in eventTable | 715
Deactivate a Row in eventTable | 715

Configure RMON History Sampling | 715

Configure RMON History Sampling Collection | **715**

View and Clear RMON History Statistics | **716**

Monitor Network Service Quality by using RMON | 717

RMON for Monitoring Service Quality | **718**

Understanding Measurement Points, Key Performance Indicators, and Baseline Values | **723**

Define and Measure Network Availability | **725**

Measure Health | **733**

Measure Performance | **742**

Health Monitoring with SNMP | 750

Health Monitoring Overview | **751**

Configure Health Monitoring on Devices Running Junos OS | **752**

Configure Health Monitoring | **756**

5 **Accounting Options**

Accounting Options Overview | 760

Configure Accounting Options, Source Class Usage and Destination Class Usage Options | 761

Configuration Statements at the [edit accounting-options] Hierarchy Level | **761**

Accounting Options Configuration | **763**

Configure Accounting-Data Log Files | **773**

- Configure How Long Backup Files Are Retained | **774**
- Configure the Maximum Size of the File | **775**
- Configure Archive Sites for the Files | **775**
- Configure Local Backup for Accounting Files | **776**
- Configure Files to Be Compressed | **776**
- Configure the Maximum Number of Files | **777**
- Configure the Storage Location of the File | **777**
- Configure Files to Be Saved After a Change in Primary Role | **777**
- Configure the Start Time for File Transfer | **778**
- Configure the Transfer Interval of the File | **778**

Manage Accounting Files | 779

Configure the Interface Profile | 780

- Configure Fields | 781
- Configure the File Information | 781
- Configure Cleared Statistics to be Reported in the Flat File | 781
- Configure the Interval | 782
- Example: Configure the Interface Profile | 782

Configure the Filter Profile | 784

- Configure the Counters | 785
- Configure the File Information | 785
- Configure the Interval | 785

Example: Configure a Filter Profile | 786

Example: Configure Interface-Specific Firewall Counters and Filter Profiles | 787

Configure Class Usage Profiles | 789

- Configure a Class Usage Profile | 789
- Configure the File Information | 790
- Configure the Interval | 790
- Create a Class Usage Profile to Collect Source Class Usage Statistics | 790
- Create a Class Usage Profile to Collect Destination Class Usage Statistics | 791

Configure the MIB Profile | 792

- Configure the File Information | 793
- Configure the Interval | 793
- Configure the MIB Operation | 793
- Configure MIB Object Names | 794
- Example: Configure a MIB Profile | 794

Configure the Routing Engine Profile | 795

- Configure Fields | 795
- Configure the File Information | 796
- Configure the Interval | 796
- Example: Configure a Routing Engine Profile | 796

6 **Monitoring Options**

Interface Alarms | 799

Alarm Overview | **799**

Example: Configure Interface Alarms | **806**

- Requirements | **806**
- Overview | **807**
- Configuration | **807**
- Verification | **810**

IP Monitoring | 811

IP Monitoring Overview | **811**

Example: Configure IP Monitoring on SRX Series Firewalls | **814**

- Requirements | **814**
- Overview | **814**
- Configuration | **814**
- Verification | **817**

Example: Configure IP Monitoring on SRX5000 line | **818**

- Requirements | **818**
- Overview | **818**
- Configuration | **820**
- Verification | **823**

Example: Configure Chassis Cluster Redundancy Group IP Address Monitoring | **826**

- Requirements | **826**
- Overview | **826**
- Configuration | **827**
- Verification | **829**

sFlow Monitoring Technology | 831

sFlow Technology Overview | **831**

sFlow Support on Switches | **832**

Example: Configure sFlow for EVPN-VXLAN Networks | **839**

- Requirements | **839**
- Overview and Topology | **840**
- Configuration | **841**
- Verification | **843**
  - Verify Configured sFlow Technology | **844**

sFlow Support on Routers | **845**

Example: Configure sFlow Technology to Monitor Network Traffic | **849**

- Requirements | **850**
- Topology | **850**
- Configuration | **851**
- Verification | **854**

sFlow Agent Address Assignment | **857**

Adaptive Sampling for Routers and Switches | 858

Adaptive Sampling Overview | **858**

Packet Flow Accelerator Diagnostics Software | 862

Packet Flow Accelerator Diagnostics Software and Other Utilities Overview | **862**

- External and Internal Ports and Network Interface Card Ports | **863**
- Packet Flow Accelerator Diagnostics Software Tests and Scripts | **864**
- Ikondiag Command | **865**
- Basic Functionality Tests | **866**
- Ethernet Tests and Scripts | **869**
- Stress Tests | **876**
- PTP Tests | **876**
- QFX-PFA-4Q Module LED Tests | **878**
- Packet Flow Accelerator Diagnostics Utilities | **879**
- Sample Output for Packet Accelerator Diagnostics Software | **886**

Install Ethernet and PTP Scripts | **894**

- Install Ethernet and PTP Scripts | **894**

Install Packet Flow Accelerator Diagnostics Software | **897**

- Packet Flow Accelerator Diagnostics Software Overview | **897**
- Verify That the QFX-PFA-4Q Expansion Module Is Installed | **898**
- Download the Packet Flow Diagnostics Software | **898**
- Copy the Packet Flow Diagnostics Software Package to the Switch | **899**
- Install the Packet Flow Diagnostics Software on the Switch | **899**
- Configure the Guest VM Options to Launch the Guest VM on the Host | **901**
- Verify That the Guest VM is Working | **904**
- Access the Guest VM | **905**

Verify That the FPGA Module Is Working | **907**

Validate the Connections Between QFX5100-24Q-AA Switch Network Ports and QFX-PFA-4Q Module Ports | **908**

Uninstall the Guest VM | **911**

7 **Monitoring Common Security Features**

Display Real-Time Information from Device to Host | 915

Display Real-Time Monitoring Information | **915**

Display Multicast Path Information | **918**

Monitor Security Policies | 922

Monitor Security Policy Statistics | **922**

Monitor Interfaces and Switching Functions | 923

Display Real-Time Interface Information | **924**

Monitor Interfaces | **927**

Monitor PPP | **928**

8 **Performance Management**

Network Analytics | 931

Network Analytics Overview | **931**

Understand Network Analytics Streaming Data | **942**

Understand Enhanced Analytics Local File Output | **950**

Understand Network Analytics Configuration and Status | **953**

Configure Queue and Traffic Monitoring | **954**

Configure a Local File for Network Analytics Data | **957**

Configure a Remote Collector for Streaming Analytics Data | **958**

Example: Configure Queue and traffic statsistics | **960**

Requirements | **961**

Overview | **961**

Configuration | **962**

Verification | **965**

Example: Configure Queue and Traffic Monitoring | **969**

Requirements | **969**

Overview | **970**

Configuration | **971**

Verification | **978**

9 **Port Mirroring**

Port Mirroring and Analyzers | 984

Port Mirroring and Analyzers | **984**

Understanding Port Mirroring and Analyzers | **985**

Port Mirroring and Analyzer Terms and Definitions | **987**

Instance Types | **991**

Port Mirroring and STP | **992**

Constraints and Limitations | **993**

Port Mirroring on QFX10000 Series Switches | **997**

Port Mirroring on QFabric | **998**

Port Mirroring on OCX Series Switches | **1000**

Port Mirroring on EX2300, EX3400, and EX4300 Switches | **1000**

Port Mirroring on ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, EX6200, and EX8200 Series Switches | **1005**

Port Mirroring on SRX Series Firewalls | **1011**

Understanding Layer 2 Port Mirroring | **1012**

Layer 2 Port Mirroring Properties | **1013**

Application of Layer 2 Port Mirroring Types | **1014**

Restrictions on Layer 2 Port Mirroring | **1017**

Configuring Port Mirroring and Analyzers | **1018**

Understanding Port Mirroring Analyzers | **1018**

Configuring Mirroring on EX9200 Switches to Analyze Traffic (CLI Procedure) | **1026**

Configuring an Analyzer for Local Traffic Analysis | **1027**

Configuring an Analyzer for Remote Traffic Analysis | **1028**

Configuring a Statistical Analyzer for Local Traffic Analysis | **1029**

Configuring a Statistical Analyzer for Remote Traffic Analysis | **1030**

Binding Statistical Analyzers to Ports Grouped at the FPC Level | **1032**

Configuring an Analyzer with Multiple Destinations by Using Next-Hop Groups | **1034**

Defining a Next-Hop Group for Layer 2 Mirroring | **1035**

Configuring Mirroring on EX4300 Switches to Analyze Traffic (CLI Procedure) | **1036**

Configuring an Analyzer for Local Traffic Analysis | **1037**

Configuring an Analyzer for Remote Traffic Analysis | **1038**

Configuring Port Mirroring | **1039**

Configuring Port Mirroring to Analyze Traffic (CLI Procedure) | **1041**

Configuring Port Mirroring for Local Traffic Analysis | **1042**

Configuring Port Mirroring for Remote Traffic Analysis | **1043**

Filtering the Traffic Entering an Analyzer | **1044**

Verifying Input and Output for Port Mirroring Analyzers on EX Series Switches | **1045**

Example: Configuring Port Mirroring Analyzers for Local Monitoring of Employee Resource Use | **1047**

Requirements | **1047**

Overview and Topology | **1048**

Mirroring All Employee Traffic for Local Analysis | **1049**

Verification | **1052**

Example: Configuring Port Mirroring for Remote Monitoring of Employee Resource Use | **1053**

Requirements | **1054**

Overview and Topology | **1054**

Mirroring Employee Traffic for Remote Analysis By Using a Statistical Analyzer | **1055**

Verification | **1066**

Example: Configuring Mirroring to Multiple Interfaces for Remote Monitoring of Employee Resource Use on EX9200 Switches | **1067**

Requirements | **1068**

Overview and Topology | **1068**

Mirroring All Employee Traffic to Multiple VLAN Member Interfaces for Remote Analysis | **1070**

Verification | **1077**

Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use Through a Transit Switch on EX9200 Switches | **1079**

Requirements | **1080**

Overview and Topology | **1080**

Mirroring All Employee Traffic for Remote Analysis Through a Transit Switch | **1082**

Verification | **1088**

Example: Configuring Mirroring for Local Monitoring of Employee Resource Use on EX4300 Switches | **1089**

Requirements | **1090**

Overview and Topology | **1090**

Mirroring All Employee Traffic for Local Analysis | **1091**

Mirroring Employee-to-Web Traffic for Local Analysis | **1093**

Verification | **1097**

Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use on EX4300 Switches | **1099**

Requirements | **1100**

Overview and Topology | **1100**

Mirroring All Employee Traffic for Remote Analysis | **1101**

Mirroring Employee-to-Web Traffic for Remote Analysis | **1106**

Verification | **1112**

Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use Through a Transit Switch on EX4300 Switches | **1113**

Requirements | **1114**

Overview and Topology | **1115**

Mirroring All Employee Traffic for Remote Analysis Through a Transit Switch | **1116**

Verification | **1122**

Configuring Port Mirroring Instances | **1124**

Layer 2 Port Mirroring Global Instance | **1124**

Configuring the Global Instance of Layer 2 Port Mirroring | **1124**

Layer 2 Port Mirroring Named Instances | **1127**

Defining a Named Instance of Layer 2 Port Mirroring | **1129**

Disabling Layer 2 Port Mirroring Instances | **1133**

Configuring Inline Port Mirroring | **1134**

Configuring Port Mirroring on Physical Interfaces | **1135**

Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface | **1135**

Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level | **1136**

Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level | **1137**

Examples: Layer 2 Port Mirroring at Multiple Levels of the Chassis | **1139**

Configuring Layer 2 Port Mirroring Over GRE Interface | **1141**

Example: Configuring Layer 2 Port Mirroring Over a GRE Interface | **1142**

Requirements | **1143**

Overview | **1143**

Configuration | **1144**

Verification | **1149**

Configuring Port Mirroring on Logical Interfaces | **1150**

Layer 2 Port Mirroring Firewall Filters | **1151**

Defining a Layer 2 Port-Mirroring Firewall Filter | **1153**

Configuring Protocol-Independent Firewall Filter for Port Mirroring | **1156**

Example: Mirroring Employee Web Traffic with a Firewall Filter | **1158**

Requirements | **1158**

Overview | **1159**

Configuring | **1160**

Verification | **1163**

Layer 2 Port Mirroring of PE Router or PE Switch Logical Interfaces | **1164**

Layer 2 Port Mirroring of PE Router or PE Switch Aggregated Ethernet Interfaces | **1166**

Applying Layer 2 Port Mirroring to a Logical Interface | **1166**

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain | **1169**

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance | **1171**

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VLAN | **1174**

Example: Layer 2 Port Mirroring at a Logical Interface | **1176**

Example: Layer 2 Port Mirroring for a Layer 2 VPN | **1179**

Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links | **1182**

Configuring Port Mirroring for Multiple Destinations | **1185**

Understanding Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups | **1185**

Defining a Next-Hop Group on MX Series Routers for Port Mirroring | **1186**

Example: Configuring Multiple Port Mirroring with Next-Hop Groups on M, MX and T Series Routers | **1188**

Example: Layer 2 Port Mirroring to Multiple Destinations | **1193**

Configuring Port Mirroring for Remote Destinations | **1197**

Layer 2 Port Mirroring to Remote Destination by Using Destination as VLAN | **1198**

Configuration Layer 2 Port Mirroring to a Remote VLAN | **1198**

Configuring Port Mirroring to a Remote VLAN | **1199**

Example: Configuring Layer 2 Port Mirroring to Remote VLAN | **1201**

Requirements | **1202**

Overview and Topology | **1202**

Mirroring Employee-to-Web Traffic for Remote Analysis | **1203**

Verification | **1209**

Configuring Port Mirroring Local and Remote Analysis | **1210**

Configuring Port Mirroring | **1210**

Configuring Port Mirroring for Local Analysis | **1211**

Configuring Port Mirroring for Remote Analysis | **1212**

Filtering the Traffic Entering an Analyzer | **1213**

Configuring Port Mirroring on SRX Series Firewalls | **1214**

Examples: Configuring Port Mirroring for Local Analysis | **1217**

Requirements | **1218**

Overview and Topology | **1218**

Example: Mirroring All Employee Traffic for Local Analysis | **1219**

Example: Mirroring Employee Web Traffic with a Firewall Filter | **1221**

Requirements | **1221**

Overview | **1221**

Configuring | **1221**

Verification | **1225**

Example: Configuring Port Mirroring for Remote Analysis | **1226**

Requirements | **1226**

Overview and Topology | **1227**

Mirroring All Employee Traffic for Remote Analysis | **1227**

Mirroring Employee-to-Web Traffic for Remote Analysis | **1229**

Verification | **1233**

1:N Port Mirroring to Multiple Destinations on Switches | **1234**

1:N Port Mirroring—Description and Configuration Guidelines | **1234**

Configure the Port-Mirroring Instance | **1236**

Configure the Native Analyzer | **1237**

Configure Next-Hop Groups | **1237**

Configure the Firewall Filter | **1237**

Configure the Interfaces | **1237**

Configure the VLANs | **1237**

Sample Configuration Results | **1238**

Monitoring Port Mirroring | **1238**

Displaying Layer 2 Port-Mirroring Instance Settings and Status | **1238**

Displaying Next-Hop Group Settings and Status | **1239**

Configure Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic | **1239**

Understanding Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic | **1239**

Configure a Filter with a Port-Mirroring Instance or with Global Port Mirroring | **1240**

Configure Mirroring for FTI Tunnels | **1243**

Attachment Points for Filters | **1247**

Suggestions for Enhancements to Your Packet-Filtering Configuration | 1247

Troubleshooting Port Mirroring | 1248

Troubleshooting Port Mirroring | 1248

Egress Port Mirroring with VLAN Translation | 1248

Egress Port Mirroring with Private VLANs | 1249

Troubleshooting Port Mirroring Configuration Error Messages | 1250

An Analyzer Configuration Returns a "Multiple interfaces cannot be configured as a member of Analyzer output VLAN" Error Message | 1250

10 System Log Messages

Overview of System Logging | 1254

System Log Overview | 1254

System Logging Facilities and Message Severity Levels | 1256

Default System Log Settings | 1258

Platform-Specific Default System Log Messages | 1260

Interpret Messages Generated in Standard Format | 1261

Manage Host OS System Log and Core Files | 1263

View Log Files On the Host OS System | 1264

Copy Log Files From the Host System To the Switch | 1264

View Core Files On the Host OS System | 1264

Copy Core Files From the Host System To the Switch | 1265

Clean Up Temporary Files on the Host OS | 1265

System Logging on a Single-Chassis System | 1266

Single-Chassis System Logging Configuration Overview | 1267

Junos OS System Log Configuration Statements | 1268

Junos OS Minimum System Logging Configuration | 1269

Example: Configure System Log Messages | 1270

Requirements | 1271

Overview | 1271

Configuration | 1271

Log Messages in Structured-Data Format | 1274

Specify Log File Size, Number, and Archiving Properties | **1274**

Include Priority Information in System Log Messages | **1276**

System Log Facility Codes and Numerical Codes Reported in Priority Information | **1278**

Include the Year or Millisecond in Timestamps | **1280**

Use Strings and Regular Expressions to Refine the Set of Logged Messages | **1281**

Junos System Log Regular Expression Operators for the match Statement | **1284**

Disable the System Logging of a Facility | **1285**

Examples: Configure System Logging | **1286**

Examples: Assign an Alternative Facility | **1288**

System Logging for a TX Matrix or TX Matrix Plus Router | 1289

Configuring System Logging for a TX Matrix Router | **1290**

Configuring System Logging for a TX Matrix Plus Router | **1292**

Configuring Message Forwarding to the TX Matrix Router | **1294**

Configuring Message Forwarding to the TX Matrix Plus Router | **1295**

Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Router | **1296**

Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Plus Router | **1299**

Configuring Optional Features for Forwarded Messages on a TX Matrix Router | **1302**

- Including Priority Information in Forwarded Messages | **1303**
- Adding a Text String to Forwarded Messages | **1303**
- Using Regular Expressions to Refine the Set of Forwarded Messages | **1304**

Configuring Optional Features for Forwarded Messages on a TX Matrix Plus Router | **1304**

- Including Priority Information in Forwarded Messages | **1305**
- Adding a Text String to Forwarded Messages | **1305**
- Using Regular Expressions to Refine the Set of Forwarded Messages | **1306**

Configuring System Logging Differently on Each T640 Router in a Routing Matrix | **1306**

Configuring System Logging Differently on Each T1600 or T4000 Router in a Routing Matrix | **1308**

Direct System Log Messages to a Remote Destination | 1310

Specify the Facility and Severity of Messages to Include in the Log | **1310**

Direct System Log Messages to a Log File | **1313**

Direct System Log Messages to a User Terminal | **1314**

Direct System Log Messages to the Console | **1315**

Direct System Log Messages to a Remote Machine or the Other Routing Engine | **1315**

Specify an Alternative Source Address for System Log Messages Directed to a Remote Destination | **1316**

Add a Text String to System Log Messages Directed to a Remote Destination | **1317**

Change the Alternative Facility Name for System Log Messages Directed to a Remote Destination | **1318**

Default Facilities for System Log Messages Directed to a Remote Destination | **1320**

Alternate Facilities for System Log Messages Directed to a Remote Destination | **1320**

Examples: Assign an Alternative Facility to System Log Messages Directed to a Remote Destination | **1322**

Direct Messages to a Remote Destination from the Routing Matrix Based on the TX Matrix Router | **1323**

Direct Messages to a Remote Destination from the Routing Matrix Based on a TX Matrix Plus Router | **1324**

Display System Log Files | 1325

Display a Log File from a Single-Chassis System | **1326**

Log File Sample Content | **1326**

Display a Log File from a Routing Matrix | **1328**

Display MD5 Log Files | **1329**

Configure System Logging for Security Devices | 1330

System Logging Overview for Security Devices | **1331**

Binary Format for Security Logs | **1333**

On-Box Logging and Reporting | **1334**

Monitor Reports | **1341**

- Threats Monitoring Report | **1342**
- Traffic Monitoring Report | **1350**

Configure On-Box Binary Security Log Files | **1352**

Configure Off-Box Binary Security Log Files | **1354**

Configure On-Box Protobuf Security Log Files in Event Mode | **1356**

Configure On-Box Protobuf Security Log Files in Stream Mode | **1357**

Configure Off-box Protobuf Security Log Files | **1359**

Send System Log Messages to a File | **1360**

Configure the System to Send All Log Messages Through eventd | **1361**

Configure Syslog over TLS | 1362

Control Plane Logs | **1363**

Example: Configure Syslog over TLS | **1363**

Requirements | **1363**

Overview | **1364**

Configuration | **1364**

Data Plane Logs | **1368**

Example: Configure the TLS Syslog Protocol on SRX Series Firewalls | **1368**

Requirements | **1368**

Overview | **1368**

Configuration | **1369**

Verification | **1372**

Monitor Log Messages | 1372

Monitor System Log Messages | **1373**

11 **Network Management and Troubleshooting**

Monitoring and Troubleshooting | 1376

Ping Hosts | **1376**

Monitor Traffic Through the Router or Switch | **1378**

Display Real-Time Statistics About All Interfaces on the Router or Switch | **1378**

Display Real-Time Statistics About an Interface on the Router or Switch | **1380**

Dynamic Ternary Content Addressable Memory Overview | **1382**

Troubleshooting DNS Name Resolution in Logical System Security Policies (Primary Administrators Only) | **1397**

Troubleshooting the Link Services Interface | **1399**

- Determine Which CoS Components Are Applied to the Constituent Links | **1399**
- Determine What Causes Jitter and Latency on the Multilink Bundle | **1402**
- Determine If LFI and Load Balancing Are Working Correctly | **1403**
- Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device | **1412**

Troubleshooting Security Policies | **1412**

- Synchronizing Policies Between Routing Engine and Packet Forwarding Engine | **1413**
- Checking a Security Policy Commit Failure | **1414**
- Verifying a Security Policy Commit | **1414**
- Debugging Policy Lookup | **1415**

Log Error Messages used for Troubleshooting ISSU-Related Problems | **1416**

- Chassisd Process Errors | **1417**
- Understanding Common Error Handling for ISSU | **1417**
- ISSU Support-Related Errors | **1421**
- Initial Validation Checks Failure | **1421**
- Installation-Related Errors | **1423**
- Redundancy Group Failover Errors | **1424**
- Kernel State Synchronization Errors | **1425**

Troubleshooting System Performance with Resource Monitoring Methodology | 1426

Resource Monitoring Usage Computation Overview | **1426**

Diagnosing and Debugging System Performance by Configuring Memory Resource Usage Monitoring on MX Series Routers | **1429**

Troubleshooting the Mismatch of jnxNatObjects Values for MS-DPC and MS-MIC | **1432**

Managed Objects for Ukernel Memory for a Packet Forwarding Engine in an FPC Slot | **1434**

Managed Objects for Packet Forwarding Engine Memory Statistics Data | **1435**

Managed Objects for Next-Hop, Jtree, and Firewall Filter Memory for a Packet Forwarding Engine in an FPC Slot | **1435**

jnxPfeMemoryErrorsTable | **1436**

pfeMemoryErrors | **1437**

Configuring Data Path Debugging and Trace Options | 1437

Understanding Data Path Debugging for SRX Series Devices | **1438**

Packet Capture from Operational Mode | **1439**

Understanding Security Debugging Using Trace Options | **1440**

Understanding Flow Debugging Using Trace Options | **1440**

Debugging the Data Path (CLI Procedure) | **1440**

Setting Flow Debugging Trace Options (CLI Procedure) | **1441**

Setting Security Trace Options (CLI Procedure) | **1442**

Displaying Log and Trace Files | **1444**

Displaying Output for Security Trace Options | **1444**

Displaying Multicast Trace Operations | **1446**

Displaying a List of Devices | **1447**

Example: Configuring End-to-End Debugging on SRX Series Device | **1449**

- Requirements | **1449**
- Overview | **1450**
- Configuration | **1450**
- Enabling Data Path Debugging | **1453**
- Verification | **1454**

Using MPLS to Diagnose LSPs, VPNs, and Layer 2 Circuits | 1455

MPLS Connection Checking Overview | **1456**

Using Packet Capture to Analyze Network Traffic | 1459

Packet Capture Overview | **1460**

Example: Enable Packet Capture on a Device | **1463**

- Requirements | **1463**
- Overview | **1463**
- Configuration | **1464**
- Verification | **1466**

Example: Configure Packet Capture on an Interface | **1468**

- Requirements | **1469**
- Overview | **1469**

Configuration | **1469**

Verification | **1470**

Example: Configure a Firewall Filter for Packet Capture | **1471**

Requirements | **1471**

Overview | **1471**

Configuration | **1472**

Verification | **1474**

Example: Configure Packet Capture for Datapath Debugging | **1474**

Requirements | **1475**

Overview | **1475**

Configuration | **1475**

Verification | **1478**

Disable Packet Capture | **1479**

Modify Encapsulation on Interfaces with Packet Capture Configured | **1479**

Delete Packet Capture Files | **1481**

Display Packet Headers | **1482**

Troubleshooting Security Devices | 1488

Troubleshooting DNS Name Resolution in Logical System Security Policies (Primary Administrators Only) | **1489**

Troubleshooting the Link Services Interface | **1490**

Determine Which CoS Components Are Applied to the Constituent Links | **1490**

Determine What Causes Jitter and Latency on the Multilink Bundle | **1493**

Determine If LFI and Load Balancing Are Working Correctly | **1494**

Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device | **1503**

Troubleshooting Security Policies | **1503**

Synchronizing Policies Between Routing Engine and Packet Forwarding Engine | **1504**

Checking a Security Policy Commit Failure | **1505**

Verifying a Security Policy Commit | **1505**

Debugging Policy Lookup | **1506**

12 **Configuration Statements and Operational Commands**

Junos CLI Reference Overview | 1509

About This Guide

Use this guide to implement and configure the network management technologies that Junos OS supports: Simple Network Management Protocol (SNMP), Remote Monitoring (RMON), Destination Class Usage (DCU) and Source Class Usage (SCU) data, and Accounting Profiles. Alarms, events, and security features are included, as is information on performance management, port mirroring and analyzers, and system logging.

RELATED DOCUMENTATION

- SNMP MIB Explorer

PART 1

Overview

Device Management Functions in Junos OS | 2

Device and Network Management Features | 5

Tracing and Logging Operations | 10

Junos Space Support for Network Management | 12

Diagnostic Tools Overview | 13

Device Management Functions in Junos OS

SUMMARY

This section provides an overview of the Junos OS (operating system).

After installing 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® operating system (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 1 on page 3.

Table 1: 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 Explorer. • SNMP MIBs—For more information about SNMP MIBs supported by Junos OS, see the "SNMP MIBs Supported by Junos OS and Junos OS Evolved" on page 575. • Standard SNMP traps—For more information about standard SNMP traps, see the "SNMP MIBs Supported by Junos OS and Junos OS Evolved" on page 575. • Enterprise-specific SNMP traps—For more information about enterprise-specific traps, see "Enterprise-Specific SNMP Traps Supported by Junos OS". • System log messages— For more information about how to view system log messages, see the System Log Explorer.
Configuration management	• Configure router attributes using the command-line interface (CLI), the Junos XML management protocol, and the NETCONF XML management protocol. For more information about configuring the router using the APIs, see the *Junos XML Management Protocol Guide* and *NETCONF XML Management Protocol Guide*. • Configuration Management MIB—For more information about the Configuration Management MIB, see Configuration Management MIB.

Table 1: Device Management Features in Junos OS *(Continued)*

Task	Junos OS Feature
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" on page 763. • 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. • Use per-ATM virtual circuit (VC) counters, available in the enterprise-specific ATM MIB. For more information about the ATM MIB, see ATM MIB. • Group source and destination prefixes into source classes and destination classes and count packets for those classes. Collect destination class and source class usage statistics. For more information about classes, see "Destination Class Usage MIB" and "Source Class Usage MIB", "Configuring Class Usage Profiles" on page 789, the *Junos OS Network Interfaces Library for Routing Devices*. • Count packets as part of a *firewall filter*. For more information about firewall filter policies, see "Enterprise-Specific SNMP MIBs Supported by Junos OS" on page 622. • Sample traffic, collect the samples, and send the collection to a host running the CAIDA cflowd utility.
Performance management	You can monitor performance in the following ways: • Use operational mode commands. For more information about monitoring performance using operational mode commands, see the CLI Explorer. • Use firewall filter. • Sample traffic, collect the samples, and send the samples to a host running the CAIDA cflowd utility. • Use the enterprise-specific Class-of-Service MIB. For more information about this MIB, see Class-of-Service MIB.

Table 1: 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. • Control access to the router using SNMPv3 and SNMP over IPv6. For more information, see "Configure Local Engine ID on SNMPv3" on page 532 and "Tracing SNMP Activity on a Device Running Junos OS" on page 518.

Device and Network Management Features

Juniper devices support features that allow you to manage the system performance, fault monitoring, and remote access.

You can use CLI operational mode commands to monitor the system health and performance of your network. Monitoring tools and commands display the current state of the device. You can filter the output to a file. Diagnostic tools and commands test the connectivity and reachability of hosts in the network.

This topic describes the functions available. To use the CLI operational tools, you must have the appropriate access privileges.

Table 2 on page 5 lists the network management features.

Table 2: Device and Network Management Features on the QFX Series, OCX Series, and EX4600 Series

Feature	Typical Uses	Documentation
Alarms and LEDs on the switch—Display status of hardware components and indicate warning or error conditions.	Fault management	*Chassis Alarm Messages on a QFX3500 Device*

Table 2: Device and Network Management Features on the QFX Series, OCX Series, and EX4600 Series *(Continued)*

Feature	Typical Uses	Documentation
Firewall filters—Control the packets that are sent to and from the network, balance network traffic, and optimize performance.	Performance management	• Routing Policies, Firewall Filters, and Traffic Policers User Guide • *Overview of Firewall Filters (QFX Series)*
In-band management—Enables connection to the switch using the same interfaces through which customer traffic flows. Communication between the switch and a remote console is enabled using SSH and Telnet services. SSH provides secure encrypted communications, whereas Telnet provides unencrypted, and therefore less secure, access to the switch.	Remote access management	• Configuring SSH Service for Remote Access to the Router or Switch • Configuring Telnet Service for Remote Access to a Router or Switch
Juniper Networks Junos OS automation scripts—Configuration and operation automation tools provided by Junos OS include commit scripts, operation scripts, event scripts, and event policies. Commit scripts enforce custom configuration rules, whereas operation scripts, event policies, and event scripts automate network troubleshooting and management.	• Configuration management • Performance management • Fault management	Automation Scripting User Guide

Table 2: Device and Network Management Features on the QFX Series, OCX Series, and EX4600 Series *(Continued)*

Feature	Typical Uses	Documentation
Junos OS command-line interface (CLI) — CLI configuration statements enable you to configure the switch based on your networking requirements, such as security, service, and performance.	• Configuration management • Performance management • User access management • Remote access management	CLI User Guide for Junos OS
Junos Space software—Multipurpose GUI-based network management system includes a base platform, the Network Application Platform, and other optional applications such as Ethernet Design, Service Now, Service Insight, and Virtual Control. **NOTE**: Junos Space does not support the OCX Series.	• Configuration management • Performance management • Fault management	Junos Space Support for Network Management
Junos XML API—XML representation of Junos OS configuration statements and operational mode commands. The Junos XML API also includes tag elements that are the counterpart to Junos CLI configuration statements.	• Configuration management • Performance management • Fault management	*Junos XML API Overview*

Table 2: Device and Network Management Features on the QFX Series, OCX Series, and EX4600 Series *(Continued)*

Feature	Typical Uses	Documentation
NETCONF XML management protocol—XML-based management protocol that client applications use to request and change configuration information on routing, switching, and security platforms running Junos OS. The NETCONF XML management protocol defines basic operations that are equivalent to Junos OS CLI configuration mode commands. Client applications use the protocol operations to display, edit, and commit configuration statements (among other operations), as administrators use CLI configuration mode commands such as `show`, `set`, and `commit` to perform those operations.	• Configuration management • Performance management • Fault management	NETCONF XML Management Protocol Developer Guide
Operational mode commands: • Monitor switch performance. For example, the `show chassis routing-engine` command shows the CPU utilization of the Routing Engine. High CPU utilization of the Routing Engine can affect performance of the switch. • View current activity and status of the device or network. For example, you can use the `ping` command to monitor and diagnose connectivity problems, and the `traceroute` command to locate points of failure on the network.	• Performance management • Fault management	CLI Explorer

Table 2: Device and Network Management Features on the QFX Series, OCX Series, and EX4600 Series *(Continued)*

Feature	Typical Uses	Documentation
Out-of-band management—Enables connection to the switch through a management interface. Out-of-band management is supported on two dedicated management Ethernet interfaces as well as on the console and auxiliary ports. The management Ethernet interfaces connect directly to the Routing Engine. Transit traffic is not allowed through the interfaces, which ensures the congestion or failures in the transit network do not affect the management of the switch.	Remote access management	• *Connect a Device to a Network for Out-of-Band Management* • *Connecting a QFX Series Device to a Management Console*
SNMP Configuration Management MIB—Provides notification for configuration changes in the form of 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. History of the last 32 configuration changes is placed in jnxCmChgEventTable.	Configuration management	SNMP MIB Explorer
SNMP MIBs and traps—Enable the monitoring of network devices from a central location. Use SNMP requests such as `get` and `walk` to monitor and view system activity. The QFX3500 switch supports SNMP Version 1 (v1), v2, and v3, and both standard and Juniper Networks enterprise-specific MIBs and traps.	Fault management	• SNMP MIB Explorer • "Understand SNMP Implementation in Junos OS" on page 371

Table 2: Device and Network Management Features on the QFX Series, OCX Series, and EX4600 Series *(Continued)*

Feature	Typical Uses	Documentation
System log messages—Log details of system and user events, including errors. You can specify the severity and type of system log messages you wish to view or save, and configure the output to be sent to local or remote hosts.	• Fault management • User access management	• System Log Explorer • "Overview of System Logging" on page 1254 • "Single-Chassis System Logging Configuration Overview" on page 1267

Tracing and Logging Operations

Tracing and logging operations enable you to track events that occur in the switch—both normal operations and error conditions—and to track the packets that are generated by or passed through the switch. The results of tracing and logging operations are placed in **/var/log** directory on the switch.

The Junos OS supports remote tracing for the following processes:

- chassisd—Chassis-control process
- eventd—Event-processing process
- cosd—Class-of-service process

You configure remote tracing using the `tracing` statement at the `[edit system]` hierarchy level.

NOTE: The `tracing` statement is not supported on the QFX3000 QFabric system.

You can disable remote tracing for specific processes on the switch using the `no-remote-trace` statement at the `[edit` *`process-name`* `traceoptions]` hierarchy level.

Logging operations use system logging mechanism similar to the UNIX syslogd utility to record systemwide, high-level operations, such as interfaces going up or down and users logging in to or out of the switch. You configure these operations by using the `syslog` statement at the `[edit system]` hierarchy level and by using the `options` statement at the `[edit ethernet-switching-options]` hierarchy level.

Tracing operations record more detailed information about the operations of the switch, including packet forwarding and routing information. You can configure tracing operations using the `traceoptions` statement.

NOTE: The `traceoptions` statement is not supported on the QFX3000 QFabric system.

You can define tracing operations in different portions of the switch configuration:

- SNMP agent activity tracing operations—Define tracing of the activities of SNMP agents on the switch. You can configure SNMP agent activity tracing operations at the `[edit snmp]` hierarchy level.
- Global switching tracing operations—Define tracing for all switching operations. You configure global switching tracing operations at the `[edit ethernet-switching-options]` hierarchy level.
- Protocol-specific tracing operations—Define tracing for a specific routing protocol. You configure protocol-specific tracing operations in the `[edit protocols]` hierarchy. Protocol-specific tracing operations override any equivalent operations that you specify in the global `traceoptions` statement.
- Tracing operations within individual routing protocol entities—Some protocols allow you to define more granular tracing operations. For example, in Border Gateway Protocol (BGP), you can configure peer-specific tracing operations. These operations override any equivalent BGP-wide operations. If you do not specify any peer-specific tracing operations, the peers inherit, first, all the BGP-wide tracing operations and, second, the global tracing operations.
- Interface tracing operations—Define tracing for individual interfaces and for the interface process itself. You define interface tracing operations at the `[edit interfaces]` hierarchy level.
- Remote tracing—To enable system-wide remote tracing, configure the `destination-override syslog host` statement at the `[edit system tracing]` hierarchy level. This specifies the remote host running the system log process (syslogd), which collects the traces. Traces are written to files on the remote host in accordance with the syslogd configuration in **/etc/syslog.conf**. By default, remote tracing is not configured.

  To override the system-wide remote tracing configuration for a particular process, include the `no-remote-trace` statement at the `[edit` *`process-name`* `traceoptions]` hierarchy. When `no-remote-trace` is enabled, the process does local tracing.

  To collect traces, use the local0 facility as the selector in the **/etc/syslog.conf** file on the remote host. To separate traces from various processes into different files, include the process name or trace-file name (if it is specified at the [edit *process-name* `traceoptions file`] hierarchy level) in the Program field in the **/etc/syslog.conf** file. If the system log server supports parsing hostname and program name, then you can separate traces from the various processes.

NOTE: During a commit check, warnings about the `traceoptions` configuration (for example, mismatch in trace file sizes or number of trace files) are not displayed on the console. However, these warnings are logged in the system log messages when the new configuration is committed.

Junos Space Support for Network Management

IN THIS SECTION

- Preparing the Device for Junos Space Management | **12**

The Juniper Networks Junos Space application, running on a Junos Space Virtual Appliance, is a comprehensive platform for building and deploying applications. This supports for collaboration, productivity, and network infrastructure and operations management. Junos Space provides a runtime environment implemented as a fabric of virtual and physical appliances.

Preparing the Device for Junos Space Management

Prerequisites

Ensure that the configuration on the QFX Series device meets the following requirements for device discovery in Junos Space:

- The device configuration has a static management IP address that is reachable from the Junos Space server.
- There is a user with full administrative privileges for Junos Space administration.
- SNMP is enabled (only if you plan on using SNMP as part of the device discovery).
- In Junos Space, set up a default device management interface (DMI) schema for the QFX Series device.

To prepare the device before using Junos Space:

1. Perform the initial configuration of the device through the console port using the Junos OS CLI. This task includes the configuration of a static management IP address and a user with root administrative privileges.

   For the QFX3500 switch, see Configuring a QFX3500 Device as a Standalone Switch.

   For the QFabric system, see *QFabric System Initial and Default Configuration Information* and *Performing the QFabric System Initial Setup on a QFX3100 Director Group*.

2. (Optional) Configure SNMP if you plan on using SNMP to probe devices during device discovery.

3. (Optional) Enable SSH if you wish to use the Secure Console feature in Junos Space.

   See Connecting to a Device by Using Secure Console.

4. In Junos Space, set up a default DMI schema. For more information about managing DMI schemas, see:

   Setting a Default DMI Schema.

RELATED DOCUMENTATION

Junos Space Network Management Platform

Diagnostic Tools Overview

IN THIS SECTION

- J-Web Diagnostic Tools | 14
- CLI Diagnostic Commands | 15

Juniper Networks devices support a suite of J-Web tools and CLI operational mode commands for evaluating system health and performance. Diagnostic tools and commands test the connectivity and reachability of hosts in the network.

- Use the J-Web Diagnose options to diagnose a device. J-Web results appear in the browser.

- Use CLI operational mode commands to diagnose a device. You can view the CLI command output on the console or management device. You can filter the output to a file.

To use the J-Web user interface and CLI operational tools, you must have the appropriate access privileges.

This section contains the following topics:

J-Web Diagnostic Tools

The J-Web diagnostic tools consist of the options that appear when you select **Troubleshoot** and **Maintain** in the task bar. Table 3 on page 14 describes the functions of the Troubleshoot options.

Table 3: J-Web Interface Troubleshoot Options

Option	Function
Troubleshoot Options	
Ping Host	Allows you to ping a remote host. You can configure advanced options for the ping operation.
Ping MPLS	Allows you to ping an MPLS endpoint using various options.
Traceroute	Allows you to trace a route between the device and a remote host. You can configure advanced options for the traceroute operation.
Packet Capture	Allows you to capture and analyze router control traffic.
Maintain Options	
Files	Allows you to manage log, temporary, and core files on the device.
Upgrade	Allows you to upgrade and manage Junos OS packages.
Licenses	Displays the summary of the licenses needed and used for each feature that requires a license. Allows you to add licenses.

Table 3: J-Web Interface Troubleshoot Options *(Continued)*

Option	Function
Reboot	Allows you to reboot the device at a specified time.

CLI Diagnostic Commands

The CLI commands available in operational mode allow you to perform the same monitoring, troubleshooting, and management tasks you can perform with the J-Web user interface. Instead of invoking the tools through a graphical interface, you use operational mode commands to perform the tasks.

CLI command output appears on the screen of your console or management device, or you can filter the output to a file. For operational commands that display output, such as the `show` commands, you can redirect the output into a filter or a file. When you display help about these commands, one of the options listed is `|`, called a *pipe*, which allows you to filter the command output.

You can use the `mtrace` command to display trace information about a multicast path from a source to a receiver.

To view a list of top-level operational mode commands, type a question mark (?) at the command-line prompt.

You can view CLI diagnostic commands at the top level of operational mode listed in Table 4 on page 15.

Table 4: CLI Diagnostic Command Summary

Command	Function
Controlling the CLI Environment	
`set` *option*	Configures the CLI display.
Diagnosis and Troubleshooting	
`clear`	Clears statistics and protocol database information.

Table 4: CLI Diagnostic Command Summary *(Continued)*

Command	Function
`mtrace`	Traces information about multicast paths from source to receiver.
`monitor`	Performs real-time debugging of various Junos OS components, including the routing protocols and interfaces.
`ping`	Determines the reachability of a remote network host.
`ping mpls`	Determines the reachability of an MPLS endpoint using various options.
`test`	Tests the configuration and application of policy filters and AS path regular expressions.
`traceroute`	Traces the route to a remote network host.
Connecting to Other Network Systems	
`ssh`	Opens secure shell connections.
`telnet`	Opens Telnet sessions to other hosts on the network.
Management	
`copy`	Copies files from one location on the device to another, from the device to a remote system, or from a remote system to the device.
`restart` *option*	Restarts the various system processes, including the routing protocol, interface, and SNMP processes.
`request`	Performs system-level operations, including stopping and rebooting the device and loading Junos OS images.
`start`	Exits the CLI and starts a UNIX shell.

Table 4: CLI Diagnostic Command Summary *(Continued)*

Command	Function
`configuration`	Enters configuration mode.
`quit`	Exits the CLI and returns to the UNIX shell.

PART 2

Operation, Administration, and Management Features

Ethernet OAM and Connectivity Fault Management for Routers | 19

Link Fault Management for Routers | 132

Ethernet OAM Link Fault Management for Switches | 167

Ethernet OAM Connectivity Fault Management for Switches | 179

Ethernet Frame Delay | 196

Ethernet Service OAM (ITU-TY.1731) for Routers | 204

CHAPTER 1

Ethernet OAM and Connectivity Fault Management for Routers

IN THIS CHAPTER

- Introduction to OAM Connectivity Fault Management (CFM) | 19
- Configure Connectivity Fault Management (CFM) | 26
- CFM Action Profile | 55
- Ethernet Local Management Interface | 62
- CFM Support for CCC Encapsulated Packets | 73
- Configure Unified ISSU for 802.1ag CFM | 75
- CFM Monitoring between CE and PE Devices | 79
- Configure Continuity Check Messages | 107
- Example: Configure Ethernet CFM on Physical Interfaces | 114
- Example: Configure Ethernet CFM on Bridge Connections | 117
- Example: Configure Ethernet CFM over VPLS | 122

Introduction to OAM Connectivity Fault Management (CFM)

SUMMARY

This section describes the Operation, Administration, and Management (OAM) of connectivity fault management (CFM).

IN THIS SECTION

- Ethernet OAM Connectivity Fault Management | 20
- IEEE 802.1ag OAM Connectivity Fault Management | 21

Ethernet OAM Connectivity Fault Management

The connectivity fault management (CFM) is defined in IEEE 802.1ag. This topic emphasizes the use of CFM in a Metro Ethernet environment.

The major features of CFM are:

- Fault monitoring using the continuity check protocol. This protocol serves as a neighbor discovery and health check protocol that identifies and maintains adjacencies at the VLAN or link level.
- Path discovery and fault verification using the linktrace protocol. Similar to IP traceroute, this protocol maps the path taken to a destination MAC address through one or more bridged networks between the source and destination.
- Fault isolation using the loopback protocol. Similar to IP ping, this protocol works with the continuity check protocol during troubleshooting.

CFM divides the service network into different administrative domains, such as operators, providers, and customers. These domains might belong to separate administrative domains.

Every administrative domain is linked with one maintenance domain that contains sufficient information for self-management, enable end-to-end monitoring, and prevent security breaches. Each maintenance domain is associated with a maintenance domain level ranging from 0 to 7, based on the network hierarchy. The outermost domains are allocated a higher level than the innermost domains. Customer end points have the highest maintenance domain level.

Each service instance in a CFM maintenance domain is called a *maintenance association*. A *maintenance association* consists of a full mesh of maintenance endpoints (MEPs) that share similar characteristics. MEPs are active CFM entities that generate and respond to CFM protocol messages.

There is also a maintenance intermediate point (MIP), which is a CFM entity similar to the MEP. However, MIP is relatively passive and only responds to CFM messages.

MEPs can be *up MEPs* or *down MEPs*. A link can connect a MEP at level 5 to a MEP at level 7. The interface at level 5 is an up MEP (because the other end of the link is at MEP level 7), and the interface at level 7 is a down MEP (because the other end of the link is at MEP level 5).

In a Metro Ethernet network, CFM is commonly used at two levels:

- By the service provider to check the connectivity among its provider edge (PE) routers
- By the customer to check the connectivity among its customer edge (CE) routers

> **NOTE**: The configured customer CFM level must be greater than service provider CFM level.

In many Metro Ethernet networks, CFM is used to monitor connectivity over a VPLS and bridge network.

NOTE: In ACX Series routers, OAM for VPLS is supported only on ACX5048, ACX5096, and ACX5448 routers and OAM for EVPN is supported only on ACX5448 and ACX710 routers.

CFM support on PTX10001-36MR, PTX10004, PTX10008, and PTX10016 devices includes the following limitations:

- Maintenance End Point (MEP) and maintenance Intermediate Point (MIP) related limitations—You cannot configure:
  - Up MEP and down MEP at same level on an interface.
- If the up MEP is higher than the down MEP, the system does not drop the CCM PDUs selectively and allows them to pass through without interruption.
- DM related timestamping on AE with child links across multiple PFEs is not supported.
- CFM packets take on the default queue. There is no forwarding class to queue (fc-to-queue) mapping, in the following instances:
  - Egress traffic, if cos-rewrite is not configured
  - Untagged traffic
- The configuration of `vlan-id-list` on OAM enabled IFLs can impact CFM scaling.
- CFM packets that are host-bound and host-generated, don't bypass the configured firewall filters for both ingress and egress direction.

IEEE 802.1ag OAM Connectivity Fault Management

IN THIS SECTION

- Connectivity Fault Management Key Elements | 23
- Best Practices for Configuring 802.1ag Ethernet OAM for VPLS | 25

Junos OS supports IEEE 802.1ag connectivity fault management. Ethernet interfaces on M7i and M10i routers with the Enhanced CFEB (CFEB-E) and on M120, M320, MX Series, T Series, and PTX Series routers support the IEEE 802.1ag standard for Operation, Administration, and Management (OAM). The

IEEE 802.1ag standard facilitates Ethernet connectivity fault management (CFM) that helps to monitor an Ethernet network comprising one or more service instances.

In Junos OS Release 9.3 and later, CFM also supports aggregated Ethernet interfaces. CFM sessions operate in distributed mode on the Flexible PIC Concentrator (FPC) on aggregated Ethernet interfaces. As a result, graceful Routing Engine switchover (GRES) is supported on aggregated Ethernet interfaces. In releases before Junos OS Release 13.3, CFM sessions operate in centralized mode on the Routing Engine. However, CFM sessions are not supported on aggregated Ethernet interfaces if the interfaces that form the aggregated Ethernet bundle are in mixed mode. Additionally, CFM sessions with a continuity check message (CCM) interval of 10 milliseconds are not supported over aggregated Ethernet interfaces.

CFM sessions are distributed by default. All CFM sessions must operate in either only distributed or only centralized mode. A mixed operation of distributed and centralized modes for CFM sessions is not supported. To disable the distribution of CFM sessions on aggregated Ethernet interfaces and make the sessions operate in centralized mode, include the `no-aggregate-delegate-processing` statement at the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level.

NOTE: As a requirement for Ethernet OAM 802.1ag to work, distributed periodic packet management (PPM) runs on the Routing Engine and Packet Forwarding Engine. You can only disable PPM on the Packet Forwarding Engine. To disable PPM on the PFE, include the `ppm no-delegate-processing` statement at the `[edit routing-options ppm]` hierarchy level.

NOTE:

- MX Series Virtual Chassis does not support distributed inline connectivity fault management.
- ACX Series routers support CFM on aggregated Ethernet interfaces with continuity check interval of 100 milliseconds or higher.
- CFM sessions are supported on aggregated Ethernet interfaces if the interfaces that form the aggregated Ethernet bundle are in mixed mode when the `no-aggregate-delegate-processing` command is enabled.
- Starting in Junos OS Release 14.2, for CFM sessions in centralized mode, we recommend that you configure a maximum of 40 CFM sessions with continuity check message (CCM) interval of 100 milliseconds (100 ms) or a maximum of 400 CFM sessions with CCM interval of 1 second (1 s). If CFM sessions are configured beyond this limit, CFM might not work as expected. You might observe issues when the state of multiple links change or when the line cards are restarted.

> Note that these limits have been derived by considering a protocol data unit (PDU) load of 400 packets per second (pps) on the Routing Engine. This limit varies depending on the Routing Engine load. If the Routing Engine experiences heavy load, expect some variations to this limit.

Starting in Junos OS Release 10.3, CFM is not supported on untagged aggregated Ethernet member links on interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series routers. However, CFM is supported on both untagged and tagged aggregated Ethernet logical interfaces configured on MPCs and MICs.Starting in Junos OS Release 12.3, CFM does not support Multichassis Link Aggregation (MC-LAG). We recommend not to configure the `mc-ae` statement when you configure CFM.

Starting in Junos OS Release 11.3, on T Series and M320 routers, CFM is not supported on interfaces configured with CCC encapsulation. If you configure CFM, the system displays the following message: "`MEPs cannot be configured on ccc interface on this platform`".

Starting in Junos OS Release 17.4, you can enable support for IEEE 802.1ag CFM on pseudowire service interfaces by configuring maintenance intermediate points (MIPs) on the pseudowire service interfaces. Pseudowire service interfaces support configuring of subscriber interfaces over MPLS pseudowire termination. Termination of subscriber interfaces over PW enables network operators to extend their MPLS domain from the Access/Aggregation network to the service edge and use uniform MPLS label provisioning for a larger portion of their network.

> **NOTE**: The CFM MIP session is supported only on the pseudowire services interface and not on the pseudowire services tunnel interface.

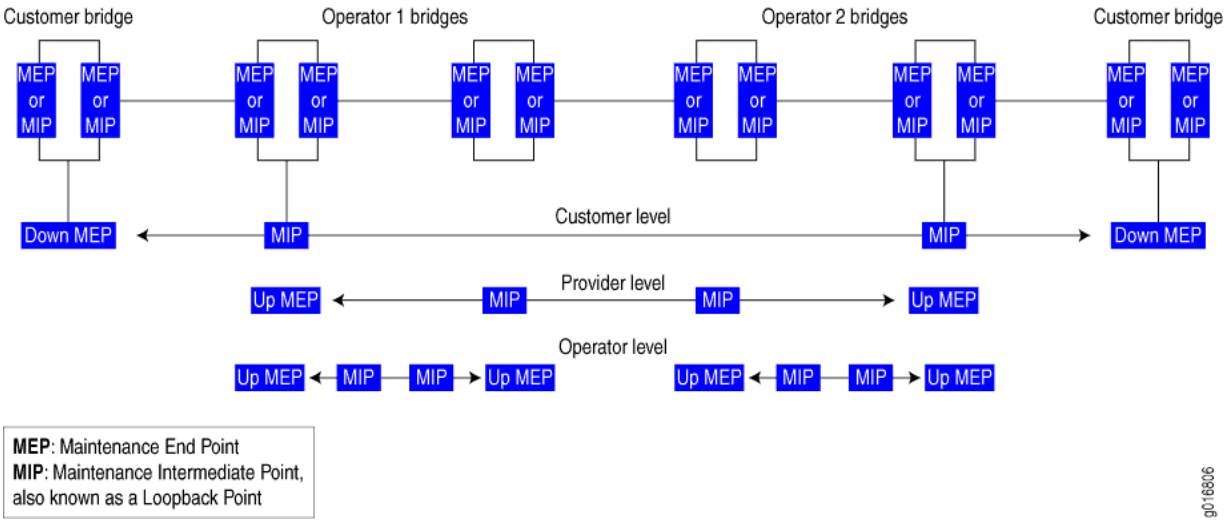
IEEE 802.1ag OAM supports *graceful Routing Engine switchover* (GRES). IEEE 802.1ag OAM is supported on untagged, single tagged, and stacked VLAN interfaces.

On EX Series switches, to use the CFM feature, you must first add the CFM to basic Junos OS by installing an enhanced feature license (EFL). See Licenses for EX Series for more details.

Connectivity Fault Management Key Elements

Figure 1 on page 24 shows the relationships among the customer, provider, and operator Ethernet bridges, maintenance domains, maintenance association end points (MEPs), and maintenance intermediate points (MIPs).

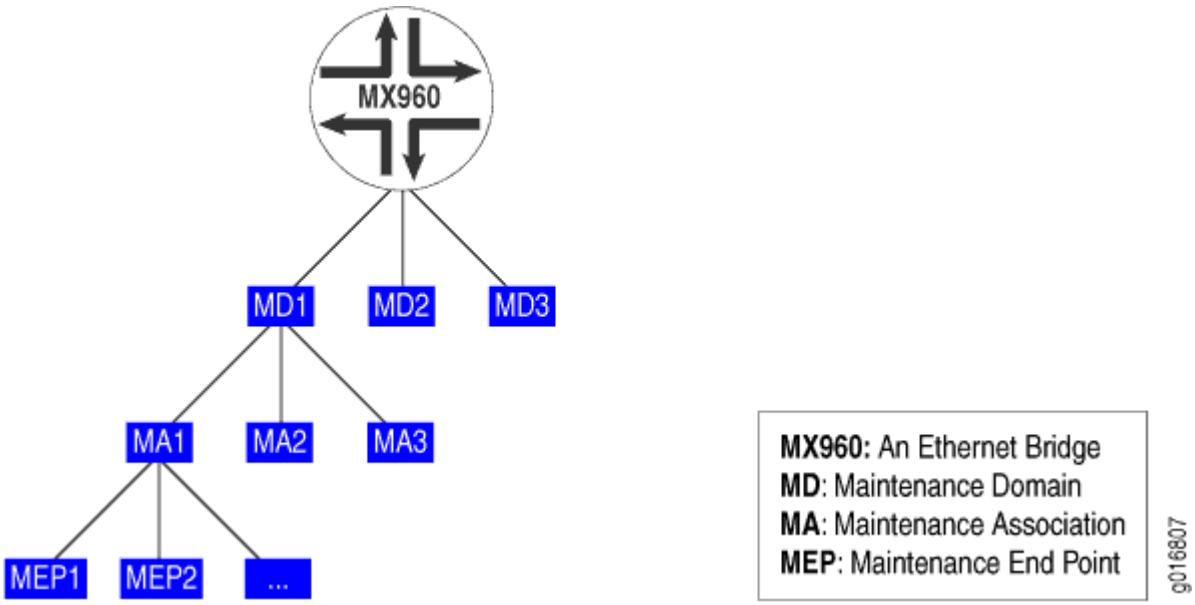
Figure 1: Relationship Among MEPs, MIPs, and Maintenance Domain Levels



NOTE: On ACX Series routers, the maintenance intermediate points (MIP) are supported only on the ACX5048 and ACX5096 routers.

A maintenance association is a set of MEPs configured with the same maintenance association identifier and maintenance domain level. Figure 2 on page 24 shows the hierarchical relationships between the Ethernet bridge, maintenance domains, maintenance associations, and MEPs.

Figure 2: Relationship Among Bridges, Maintenance Domains, Maintenance Associations, and MEPs



Best Practices for Configuring 802.1ag Ethernet OAM for VPLS

BEST PRACTICE: The logical interfaces in a VPLS routing instance may have the same or different VLAN configurations. VLAN normalization is required to switch packets correctly among these interfaces. VLAN normalization is effectively VLAN translation wherein the VLAN tags of the received packet need to be translated if they are different than the normalized VLAN tags.

For MX Series routers, the normalized VLAN is specified using one of the following configuration statements in the VPLS routing instance:

- `vlan-id` *`vlan-number`*
- `vlan-id none`
- `vlan-tags outer` *`outer-vlan-number`* `inner` *`inner-vlan-number`*

You must configure `vlan-maps` explicitly on all interfaces belonging to the routing instance.

The following forwarding path considerations must be observed:

- Packet receive path:
  - This is the forwarding path for packets received on the interfaces.
  - 802.1ag Ethernet OAM for VPLS uses implicit interface filters and forwarding table filters to flood, accept, and drop the CFM packets.
- Packet transmit path:
  - The JUNOS Software uses the router's hardware-based forwarding for CPU-generated packets.
  - For Down MEPs, the packets are transmitted on the interface on which the MEP is configured.
  - In MX series routers, for Up MEPs, the packet must be flooded to other interfaces in the VPLS routing instance. The router creates a flood route tied to a flood next hop (with all interfaces to flood) and then sources the packet to be forwarded with this flood route.
  - The router also uses implicit-based forwarding for CPU generated packets. The result is for the flood next hop tied to the flood route to be tied to the filter term. The filter term uses match criteria to correctly identify the host- generated packets.

SEE ALSO

connectivity-fault-management

Release History Table

Release	Description
17.4R1	Starting in Junos OS Release 17.4, you can enable support for IEEE 802.1ag CFM on pseudowire service interfaces by configuring maintenance intermediate points (MIPs) on the pseudowire service interfaces.
14.2	Starting in Junos OS Release 14.2, for CFM sessions in centralized mode, we recommend that you configure a maximum of 40 CFM sessions with continuity check message (CCM) interval of 100 milliseconds (100 ms) or a maximum of 400 CFM sessions with CCM interval of 1 second (1 s).
12.3	Starting in Junos OS Release 12.3, CFM does not support Multichassis Link Aggregation (MC-LAG). Do not configure the `mc-ae` statement when you configure CFM.
11.3	Starting in Junos OS Release 11.3, on T Series and M320 routers, CFM is not supported on interfaces configured with CCC encapsulation.
10.3	Starting in Junos OS Release 10.3, on interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series routers, CFM is not supported on untagged aggregated Ethernet member links. MPCs and MICs do support CFM on untagged and tagged aggregated Ethernet logical interfaces.
9.3	In Junos OS Release 9.3 and later, CFM also supports aggregated Ethernet interfaces.

Configure Connectivity Fault Management (CFM)

IN THIS SECTION

- Create a Maintenance Domain | **27**
- Create a Maintenance Association | **28**
- Configure Maintenance Intermediate Points (MIPs) | **29**
- Configure Maintenance Association Intermediate Points in ACX Series | **31**
- Configure a MEP to Generate and Respond to CFM Protocol Messages | **35**
- Configure Service Protection for VPWS over MPLS Using the MEP Interface | **39**

- Configure Linktrace Protocol in CFM | **44**
- Continuity Check Protocol Parameters Overview | **45**
- Configuring Continuity Check Protocol Parameters for Fault Detection | **46**
- Configuring Rate Limiting of Ethernet OAM Messages | **47**
- Enabling Enhanced Connectivity Fault Management Mode | **50**
- Configure Connectivity Fault Management for Interoperability During Unified In-Service Software Upgrades | **52**
- Junos OS Support for Performance Monitoring Compliant with Technical Specification MEF 36 | **53**
- Damping CFM performance Monitoring Traps and Notifications to Prevent Congestion of The NMS | **54**

Use this topic to configure connectivity fault management features such as maintenance domains, maintenance associations, maintenance intermediate points (MIPs), and continuity check parameters. You can also use this topic to configure an action profile to specify the CFM action that must be performed when a specific CFM event occurs.

Starting in Junos OS Evolved 22.4R2 Release, the connectivity fault management process (cfmd) runs only when the `ethernet connectivity-fault-management` protocol is configured.

Create a Maintenance Domain

To enable connectivity fault management (CFM) on an Ethernet interface, you must first configure a maintenance domain and specify the name of the maintenance domain. You can also specify the format of the name. For instance, if you specify the name format to be domain name service (DNS) format, you can specify the name of the maintenance domain as www.juniper.net. The default name format is ASCII character string.

NOTE: For logical interfaces, the maintenance domain name must be unique across logical systems. If you configure the same maintenance domain name across logical systems, then you receive the following error message: `error: configuration check-out failed`.

During the creation of the maintenance domain, you can also specify the maintenance domain level. The maintenance domain level indicates the nesting relationship between various maintenance domains. The maintenance domain level is embedded in each of the CFM frames.

NOTE: The configuration display entries in the CFM maintenance domain list are "ordered by system" rather than "ordered by user."

To create a maintenance domain:

1. In configuration mode, create a maintenance domain by specifying the name and the name format at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenance-domain md-name name-format option
```

NOTE: If you configure the maintenance domain name length greater than 45 octet, then the following error message is displayed: `error: configuration check-out failed`.

2. Specify the maintenance domain level by specifying the value at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenace-domain md-name level number
```

SEE ALSO

connectivity-fault-management

maintenance-domain

name-format

level

Create a Maintenance Association

To create a maintenance association, include the `maintenance-association` *ma-name* statement at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *domain-name*] hierarchy level.

Maintenance association names can be in one of the following formats:

- As a plain ASCII character string
- As the VLAN identifier of the VLAN you primarily associate with the maintenance association

- As a two-octet identifier in the range from 0 through 65,535
- As a name in the format specified by RFC 2685

The default short name format is an ASCII character string.

To configure the maintenance association short name format, include the `short-name-format (character-string | vlan | 2octet | rfc-2685-vpn-id)` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`*`]` hierarchy level.

> **NOTE**: The configuration display entries in the CFM maintenance domain list are "ordered by system" rather than "ordered by user."

SEE ALSO

> *connectivity-fault-management*
>
> *name-format*
>
> *short-name-format*

Configure Maintenance Intermediate Points (MIPs)

MX Series routers support maintenance intermediate points (MIPs) for the Ethernet OAM 802.1ag CFM protocol at a bridge-domain level. This enables you to define a maintenance domain for each default level. The MIPs names are created as `default-`*`level-number`* at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain]` hierarchy level. Use the `bridge-domain`, `instance`, `virtual-switch`, and `mip-half-function` MIP options to specify the MIP configuration.

Use the `show oam ethernet connectivity-fault-management mip (bridge-domain | instance-name | interface-name)` command to display the MIP configurations.

To configure the maintenance intermediate point (MIP):

1. Configure a bridge domain under a user-defined virtual switch by specifying the `virtual-switch` statement and the name of the user-defined virtual switch, at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `default-`*`x`*`]` hierarchy level.

NOTE: A bridge domain must be specified by name only if it is configured by including the `vlan-id` statement under the `virtual-switch` statement. If a bridge domain is configured with a range of VLAN IDs, then the VLAN IDs must be explicitly listed after the bridge domain name.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
default-x]
user@host# set virtual-switch virtual-switch-name bridge-domain bridge-domain-name vlan-id
value
```

NOTE: You can also configure the bridge domain for the default virtual switch by including the `bridge-domain` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]` hierarchy level.

2. Configure the VPLS routing instance for the default maintenance domain.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]
user@host# set instance instance-name
```

3. Configure the maintenance intermediate point (MIP) half function to divide the MIP fuctionality into two unidirectional segments to improve network coverage by increasing the number of MIPs that are monitored. The MIP half function also responds to loop-back and link-trace messages to identify faults.

NOTE: Whenever a MIP is configured and a bridge domain is mapped to multiple maintenance domains or maintenance associations, it is essential that the `mip-half-function` value for all maintenance domains and maintenance associations be the same.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
default- x]
user@host# set mip-half-function (none | default | explicit)
```

SEE ALSO

bridge-domain

connectivity-fault-management

instance

mip-half-function

virtual-switch

Configure Maintenance Association Intermediate Points in ACX Series

IN THIS SECTION

- Configure the Maintenance Domain Bridge Domain | 32
- Configure the Maintenance Domain MIP Half Function | 32
- Configure the Maintenance Association Intermediate Points with Bridge Domain | 32
- Configure the Maintenance Association Intermediate Points with Circuit Cross-Connect | 33
- Configure the Maintenance Association Intermediate Points with Bridge Domain when Maintenance Association End Point is Configured | 33
- Configure the Maintenance Intermediate Points with Circuit Cross-Connect when Maintenance Association End Point is Configured | 34

Maintenance Intermediate Point (MIP) provides monitoring capability of intermediate points for services such as Layer 2 bridging, Layer 2 circuit, and Layer 2 VPN. ACX5048 and ACX5096 routers support MIPs for the Ethernet OAM 802.1ag CFM protocol. Use the bridge-domain, interface, and mip-half-function MIP options to specify the MIP configuration.

NOTE: ACX5048 and ACX5096 routers do not support MIP configuration on VPLS services.

NOTE: ACX5448 router do not support MIP.

NOTE: Whenever a MIP is configured and a bridge domain is mapped to multiple maintenance domains or maintenance associations, it is essential that the `mip-half-function` value for all maintenance domains and maintenance associations be the same.

To display MIP configurations, use the `show oam ethernet connectivity-fault-management mip (bridge-domain | instance-name | interface-name)` command.

The following MIP configurations are supported in ACX5048 and ACX5096 routers:

- MIP with with bridge domain
- MIP with circuit cross-connect (CCC)
- MIP with bridge domain when maintenance association end point is configured
- MIP with CCC when maintenance association end point is configured

The following sections describe MIP configuration:

Configure the Maintenance Domain Bridge Domain

To configure the bridge domain, include the `vlans` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`maintenance-domain-name`*`]` hierarchy level.

> **NOTE**: The Layer 2 CLI configurations and show commands for ACX5048 and ACX5096 routers differ compared to other ACX Series routers. For more information, see Layer 2 Next Generation Mode for ACX Series.

Configure the Maintenance Domain MIP Half Function

MIP Half Function (MHF) divides MIP functionality into two unidirectional segments, improves visibility with minimal configuration, and improves network coverage by increasing the number of points that can be monitored. MHF extends monitoring capability by responding to loopback and linktrace messages to help isolate faults.

Whenever a MIP is configured and a bridge domain is mapped to multiple maintenance domains or maintenance associations, it is essential that the *MIP half function* value for all maintenance domains and maintenance associations be the same. To configure the MIP half function, include the `mip-half-function` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`maintenance-domain-name`*`]` hierarchy level.

Configure the Maintenance Association Intermediate Points with Bridge Domain

In ACX5048 and ACX5096 routers, you can configure the MIP with bridge domain. The following is a sample to configure the MIP with bridge domain:

```
[edit protocols]
oam {
    ethernet {
```

```
        connectivity-fault-management {
            maintenance-domain default-6 {
                vlan bd1;
                mip-half-function default;
            }
        }
    }
}
```

Configure the Maintenance Association Intermediate Points with Circuit Cross-Connect

In ACX5048 and ACX5096 routers, you can configure the MIP with circuit cross-connect (CCC). The following is a sample to configure the MIP with CCC:

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain default-6 {
                interface xe-0/0/42.0;
                mip-half-function default;
            }
        }
    }
}
```

Configure the Maintenance Association Intermediate Points with Bridge Domain when Maintenance Association End Point is Configured

In ACX5048 and ACX5096 routers, you can configure the MIP with bridge domain when a maintenance association end point (MEP) is configured. The following is a sample to configure the MIP with bridge domain when MEP is configured:

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain md2 {
                level 5;
                mip-half-function default;
```

```
                maintenance-association ma2 {
                    continuity-check {
                        interval 1s;
                    }
                    mep 222 {
                        interface xe-0/0/42.0;
                        direction up;
                    }
                }
            }
        }
    }
}
```

Configure the Maintenance Intermediate Points with Circuit Cross-Connect when Maintenance Association End Point is Configured

In ACX5048 and ACX5096 routers, you can configure the MIP with circuit cross-connect (CCC) when a maintenance association end point (MEP) is configured. The following is a sample to configure the MIP with CCC when MEP is configured:

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain md2 {
                level 5;
                mip-half-function default;
                maintenance-association ma2 {
                    continuity-check {
                        interval 1s;
                    }
                    mep 222 {
                        interface xe-0/0/42.0;
                        direction up;
                    }
                }
            }
        }
    }
}
```

SEE ALSO

bridge-domain

connectivity-fault-management

instance

mip-half-function

Configure a MEP to Generate and Respond to CFM Protocol Messages

IN THIS SECTION

- Configure a Maintenance Association End Point (MEP) | **36**
- Configure a Remote Maintenance Association End Point (MEP) | **38**

A maintenance association end point (MEP) refers to the boundary of a domain. A MEP generates and responds to connectivity fault management (CFM) protocol messages. You can configure multiple up MEPs for a single combination of maintenance association ID and maintenance domain ID for interfaces belonging to a particular VPLS service or a bridge domain. You can configure multiple down MEPs for a single instance of maintenance domain identifier and maintenance association name to monitor services provided by Virtual Private LAN service (VPLS), bridge domain, circuit cross-connect (CCC), or IPv4 domains.

For layer 2 VPNs routing instances (local switching) and EVPN routing instances, you can also configure multiple up MEPs for a single combination of maintenance association ID and maintenance domain ID on logical interfaces. The logical interface can be configured on different devices or on the same device. To support multiple up MEPs on two IFLs, enhanced IP network services must be configured for the chassis.

You can enable automatic discovery of a MEP. With automatic discovery a MEP is enabled to accept continuity check messages (CCMs) from all remote MEPs of the same maintenance association. If automatic discovery is not enabled, the remote MEPs must be configured. If the remote MEP is not configured, the CCMs from the remote MEP are treated as errors.

Continuity measurement is provided by an existing continuity check protocol. The continuity for every remote MEP is measured as the percentage of time that remote MEP was operationally up over the total administratively enabled time. Here, the operational uptime is the total time during which the CCM adjacency is active for a particular remote MEP and the administrative enabled time is the total time during which the local MEP is active. You can also restart the continuity measurement by clearing the currently measured operational uptime and the administrative enabled time.

Configure a Maintenance Association End Point (MEP)

To configure a maintenance association end point:

1. Specify an ID for the MEP at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`*]. You can specify any value from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name]
user@host# set mep mep-id
```

2. Enable maintenance end point automatic discovery so the MEP can accept continuity check messages (CCMs) from all remote MEPs of the same maintenance association.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set auto-discovery
```

3. Specify the direction in which the CCM packets are transmitted for the MEP. You can specify up or down. If you specify the direction as up, CCMs are transmitted out of every logical interface that is part of the same bridging or VPLS instance except for the interface configured on the MEP. If you specify the direction as down, CCMs are transmitted only out of the interface configured on the MEP.

> **NOTE**: Ports in the Spanning Tree Protocol (STP) blocking state do not block CFM packets destined to a down MEP. Ports in an STP blocking state without the continuity check protocol configured do block CFM packets.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set direction down
```

> **NOTE**: Starting with Junos OS Release 12.3, for all interfaces configured on Modular Port Concentrators (MPCs) on MX Series 5G Universal Routing Platforms, you no longer need to configure the `no-control-word` statement for all Layer 2 VPNs and Layer 2 circuits over which you are running CFM MEPs. For all other interfaces on MX Series routers and on all other routers and switches, you must continue to configure the `no-control-word` statement at the [`edit routing-instances` *`routing-instance-name`* `protocols l2vpn`] or [`edit protocols l2circuit neighbor`

neighbor-id interface *interface-name*] hierarchy level when you configure CFM MEPs. Otherwise, the CFM packets are not transmitted, and the show oam ethernet connectivity-fault-management mep-database command does not display any remote MEPs.

4. Specify the interface to which the MEP is attached. It can be a physical interface, logical interface, or trunk interface. On MX Series routers, the MEP can be attached to a specific VLAN of a trunk interface.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set interface interface-name
```

5. Specify the IEEE 802.1 priority bits that are used by continuity check and link trace messages. You can specify a value from through 7 as the priority.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set priority number
```

6. Specify the lowest priority defect that generates a fault alarm whenever CFM detects a defect. Possible values include: all -defects, err-xcon, mac-rem-err-xcon, no-defect, rem-err-xcon, and xcon.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set lowest-priority-defect mac-rem-err-xcon
```

7. Specify the ID of the remote MEP at the [edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name* mep *mep-id*]. You can specify any value from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-namemep mep-id]
user@host# set remote-mep mep-id
```

SEE ALSO

priority

Configure a Remote Maintenance Association End Point (MEP)

To configure a remote maintenance association end point:

1. Configure the remote MEP by specifying the MEP ID at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`* `mep` *`mep-id`*]. You can specify any value from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-namemep mep-id]
user@host# edit remote-mep mep-id
```

2. Specify the name of the action profile to be used for the remote MEP by including the `action-profile` *`profile-name`* statement at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`* `mep` *`mep-id`* `remote-mep` *`remote-mep-id`*]. The profile must be defined at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-namemep mep-id remote-mep remote-mep-id]
user@host# set action-profile profile-name
```

3. Configure the remote MEP to detect initial loss of connectivity. By default, the MEP does not generate loss-of-continuity (LOC) defect messages. When you configure the `detect-loc` statement, a loss-of-continuity (LOC) defect is detected if no continuity check message is received from the remote MEP within a period equal to 3.5 times the continuity check interval configured for the maintenance association. If a LOC defect is detected, a syslog error message is generated.

> **NOTE**: When you configure connectivity-fault management (CFM) along with `detect-loc`, any `action-profile` configured to bring down the interface is executed if continuity check message is not received . However, the `action-profile` is not executed if you have not configured `detect-loc` and continuity check message is not received.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-namemep mep-id remote-mep remote-mep-id]
user@host# set detect-loc
```

SEE ALSO

remote-mep

RELATED DOCUMENTATION

action-profile

auto-discovery

connectivity-fault-management

detect-loc

direction

lowest-priority-defect

Configure Service Protection for VPWS over MPLS Using the MEP Interface

You can enable service protection for a virtual private wire service (VPWS) over MPLS by specifying a working path or protect path on the MEP. Service protection provides end-to-end connection protection of the working path in the event of a failure.

To configure service protection, you must create two separate transport paths—a working path and a protect path. You can specify the working path and protect path by creating two maintenance associations. To associate the maintenance association with a path, you must configure the `interface` statement for the MEP within the maintenance association and specify the path as working or protect.

NOTE: If the path is not specified, the session monitors the active path.

Table 5 on page 39 describes the available service protection options.

Table 5: Service Protection Options

Option	Description
`working`	Specifies the working path.
`protect`	Specifies the protect path.

In this configuration, we enable service protection for the VPWS service. The CCM session is configured for the working path and references the CCM session configured for the protect path using the `protect-`

`maintenance-association` statement. The name of the protect transport path for the maintenance association is configured and associated with the maintenance association for the working path.

To configure service protection for VPWS over MPLS:

1. In configuration mode, create a maintenance domain by specifying the name and the name format at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenance-domain md-name name-format option
```

> **NOTE**: If you configure the maintenance domain name length greater than 45 octet, then the following error message is displayed: `error: configuration check-out failed`.

2. Specify the maintenance domain level by specifying the value at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenace-domain md-name level number
```

3. Create a maintenance association for the working path by specifying the name and the short name format at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *md-name*] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name]
user@host# set maintenance-association test-ma short-name-format option
```

4. Specify the maintenance association name used for connection protection and the name of the automatic-protection-switching profile (aps-profile) at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *md-name* `maintenance-association` *ma-name*] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name]
user@host# set protect-maintenance-association ma-name aps-profile aps-profile-name
```

5. Specify the time to wait between transmissions of continuity check messages at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *md-name* `maintenance-association` *ma-name* `continuity-check`] hierarchy level. The duration can be one of the following values: 10 minutes(10m),

1 minute(1m), 10 seconds(10s), 1 second(1s), 100 milliseconds(100ms), or 10 milliseconds(10ms). The default value is 1 minute.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name continuity-check]
user@host# set interval option
```

6. Specify an ID for the MEP at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`*]. You can specify any value from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name]
user@host# set mep mep-id
```

7. Enable maintenance end point automatic discovery so the MEP can accept continuity check messages (CCMs) from all remote MEPs of the same maintenance association.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set auto-discovery
```

8. Specify the direction in which the CCM packets are transmitted for the MEP. You can specify up or down. If you specify the direction as up, CCMs are transmitted out of every logical interface that is part of the same bridging or VPLS instance except for the interface configured on the MEP. If you specify the direction as down, CCMs are transmitted only out of the interface configured on the MEP.

NOTE: Ports in the Spanning Tree Protocol (STP) blocking state do not block CFM packets destined to a down MEP. Ports in an STP blocking state without the continuity check protocol configured do block CFM packets.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set direction down
```

NOTE: Starting with Junos OS Release 12.3, for all interfaces configured on Modular Port Concentrators (MPCs) on MX Series 5G Universal Routing Platforms, you no longer need to configure the `no-control-word` statement for all Layer 2 VPNs and Layer 2 circuits over which you are running CFM MEPs. For all other interfaces on MX Series routers and on all other routers and switches, you must continue to configure the `no-control-word` statement at the `[edit routing-instances routing-instance-name protocols l2vpn]` or `[edit protocols l2circuit neighbor neighbor-id interface interface-name]` hierarchy level when you configure CFM MEPs. Otherwise, the CFM packets are not transmitted, and the `show oam ethernet connectivity-fault-management mep-database` command does not display any remote MEPs.

9. Specify the interface to which the MEP is attached. It can be a physical interface, logical interface, or trunk interface. On MX Series routers, the MEP can be attached to a specific VLAN of a trunk interface. Also, specify the transport path as working.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set interface interface-name working
```

10. Create a maintenance association for the protection path by specifying the name and the short name format at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name]` hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name]
user@host# set maintenance-association ma-name short-name-format option
```

11. Specify the time to wait between transmissions of continuity check messages at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name maintenance-association ma-name continuity-check ]` hierarchy level. The duration can be one of the following values: 10 minutes(10m), 1 minute(1m), 10 seconds(10s), 1 second(1s), 100 milliseconds(100ms), or 10 milliseconds(10ms). The default value is 1 minute.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name continuity-check]
user@host# set interval option
```

12. Specify an ID for the MEP at the [edit protocols oam ethernet connectivity-fault-management maintenance-domain *domain-name* maintenance-association *ma-name*]. You can specify any value from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name]
user@host# set mep mep-id
```

13. Enable maintenance end point automatic discovery so the MEP can accept continuity check messages (CCMs) from all remote MEPs of the same maintenance association.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set auto-discovery
```

14. Specify the direction in which the CCM packets are transmitted for the MEP. You can specify up or down. If you specify the direction as up, CCMs are transmitted out of every logical interface that is part of the same bridging or VPLS instance except for the interface configured on the MEP. If you specify the direction as down, CCMs are transmitted only out of the interface configured on the MEP.

NOTE: Ports in the Spanning Tree Protocol (STP) blocking state do not block CFM packets destined to a down MEP. Ports in an STP blocking state without the continuity check protocol configured do block CFM packets.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set direction down
```

NOTE: Starting with Junos OS Release 12.3, for all interfaces configured on Modular Port Concentrators (MPCs) on MX Series 5G Universal Routing Platforms, you no longer need to configure the no-control-word statement for all Layer 2 VPNs and Layer 2 circuits over which you are running CFM MEPs. For all other interfaces on MX Series routers and on all other routers and switches, you must continue to configure the no-control-word statement at the [edit routing-instances *routing-instance-name* protocols l2vpn] or [edit protocols l2circuit neighbor *neighbor-id* interface *interface-name*] hierarchy level when you configure CFM MEPs. Otherwise, the CFM packets are not transmitted, and the show oam ethernet connectivity-fault-management mep-database command does not display any remote MEPs.

15. Specify the interface to which the MEP is attached. It can be a physical interface, logical interface, or trunk interface. On MX Series routers, the MEP can be attached to a specific VLAN of a trunk interface. Also, specify the transport path as working.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@host# set interface interface-name protect
```

SEE ALSO

- *auto-discovery*
- *interval*
- *name-format*
- *protect-maintenance-association*
- *short-name-format*

Configure Linktrace Protocol in CFM

The linktrace protocol is used for path discovery between a pair of maintenance points. Linktrace messages are triggered by an administrator using the `traceroute` command to verify the path between a pair of MEPs under the same maintenance association. Linktrace messages can also be used to verify the path between an MEP and an MIP under the same maintenance domain. The linktrace protocol enables you to configure the time to wait for a response. If no response is received for a linktrace request message, the request and response entries are deleted after the interval expires. You can also configure the number of linktrace reply entries to be stored for the corresponding linktrace request.

The operation of IEEE 802.1ag linktrace request and response messages is similar to the operation of Layer 3 `traceroute` commands. For more information about the `traceroute` command, see the Junos OS Administration Library for Routing Devices.

To configure the linktrace protocol:

1. Configure the time to wait for a linktrace response at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level. You can specify the value in minutes or seconds. The default value is 10 minutes.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set linktrace age time
```

2. Configure the number of linktrace reply entries to be stored per linktrace request. You can specify a value from 1 through 500. The default value is 100.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set linktrace path-database-sizepath-database-size
```

SEE ALSO

> *age*
>
> *path-database-size*
>
> *connectivity-fault-management*

Continuity Check Protocol Parameters Overview

The continuity check protocol is used for fault detection by maintenance end points (MEPs) within a maintenance association. The MEP periodically sends continuity check multicast messages. The continuity check protocol packets use the ethertype value 0x8902 and the multicast destination MAC address 01:80:c2:00:00:32.

The following list describes the continuity check protocol parameters you can configure:

- `interval`—Frequency of the continuity check messages (CCM) i.e time between the transmission of the CCM messages. You can specify 10 minutes (`10m`), 1 minute (`1m`), 10 seconds (`10s`), 1 second (`1s`), 100 milliseconds (`100ms`), or 10 milliseconds (`10ms`). The default value is 1 minute. For instance, if you specify the interval as 1 minute, the MEP sends the continuity check messages every minute to the receiving MEP.

  NOTE: For the continuity check message interval to be configured for 10 milliseconds, periodic packet management (PPM) runs on the Routing Engine and Packet Forwarding Engine by default. You can only disable PPM on the Packet Forwarding Engine. To disable PPM on the Packet Forwarding Engine, use the `no-delegate-processing` statement at the `[edit routing-options ppm]` hierarchy level.

  Continuity check interval of 10 milliseconds is not supported for CFM sessions over a label-switched interface (LSI).

- `hold-interval`—Frequency at which the MEP database can be flushed, if no updates occur. Receiving MEPs use the continuity check messages to build a MEP database of all MEPs in the maintenance association. The frequency is the number of minutes to wait before flushing the MEP database if no updates occur. The default value is 10 minutes.

> **NOTE**: Hold timer based flushing is applicable only for autodiscovered remote MEPs and not for statically configured remote MEPs.

The hold interval logic runs a polling timer per CFM session level (not per remote MEP level) where the polling timer duration is equal to the configured hold time. When the polling timer expires, it deletes all the autodiscovered remote MEP entries which have been in the failed state for a time period equal to or greater than the configured hold time. If the remote MEP completes the hold time duration in the failed state, then flushing will not occur until the next polling timer expires. Hence remote MEP flushing may not happen exactly at the configured hold time.

- `loss-threshold`—Number of continuity check messages that can be lost before the router marks the MEP as down. The value can be from 3 to 256 protocol data units (PDUs). The default value is 3 PDUs.

SEE ALSO

> *hold-interval*
>
> *interval*
>
> *loss-threshold*

Configuring Continuity Check Protocol Parameters for Fault Detection

The continuity check protocol is used for fault detection by a maintenance association end point (MEP) within a maintenance association. A MEP periodically generates and responds to continuity check multicast messages. The continuity check protocol packets use the ethertype value 0x8902 and the multicast destination MAC address 01:80:c2:00:00:32. The receiving MEPs use the continuity check messages (CCMs) to build a MEP database of all MEPs in the maintenance association.

To configure continuity check protocol parameters:

1. Specify the time to wait in minutes before flushing the MEP database, if no updates occur, with a value from 1 minute through 30,240 minutes. The default value is 10 minutes.

NOTE: Flushing based on the hold timer is applicable only for autodiscovered remote MEPs and not for statically configured remote MEPs.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name  continuity-check]
user@host# set hold-interval minutes
```

2. Specify the time to wait (duration) between the transmissions of CCMs. The duration can be one of the following values: 10 minutes (10m), 1 minute (1m), 10 seconds (10s), 1 second (1s), 100 milliseconds (100ms), or 10 milliseconds (10ms). The default value is 1 minute.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name continuity-check]
user@host# set interval duration
```

3. Specify the number of continuity check messages that can be lost before the router marks the MEP as down. The value can be from 3 to 256 protocol data units (PDUs). The default value is 3 PDUs.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name continuity-check]
user@host# set loss-threshold number
```

SEE ALSO

continuity-check

hold-interval

interval

loss-threshold

Configuring Rate Limiting of Ethernet OAM Messages

The M320 with Enhanced III FPC, M120, M7i, M10 with CFEB, and MX Series routers support rate limiting of Ethernet OAM messages. Depending on the connectivity fault management (CFM) configuration, CFM packets are discarded, sent to the CPU for processing, or flooded to other bridge interfaces. This feature allows the router to intercept incoming CFM packets for prevention of DoS attacks.

You can apply rate limiting of Ethernet OAM messages at either of two CFM policing levels, as follows:

- Global-level CFM policing—uses a policer at the global level to police the CFM traffic belonging to all the sessions.
- Session-level CFM policing—uses a policer created to police the CFM traffic belonging to one session.

To configure global-level CFM policing, include the `policer` statement and its options at the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level.

To configure session-level CFM policing, include the `policer` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`md-name`* `level` *`number`* `maintenance-association` *`ma-name`*`]` hierarchy level.

The following example shows a CFM policer used for rate-limiting CFM:

```
[edit]
firewall {
    policer cfm-policer {
        if-exceeding {
            bandwidth-limit 8k;
            burst-size-limit 2k;
        }
        then discard;
    }
}
```

Case 1: Global-Level CFM Policing

This example shows a global level policer, at the CFM level, for rate-limiting CFM. The `continuity-check` *`cfm-policer`* statement at the global `[edit protocols oam ethernet connectivity-fault-management policer]` hierarchy level specifies the policer to use for policing all continuity check packets of the CFM traffic belonging to all sessions. The `other` *`cfm-policer1`* statement at the `[edit protocols oam ethernet connectivity-fault-management policer]` hierarchy level specifies the policer to use for policing all non-continuity check packets of the CFM traffic belonging to all sessions. The `all` *`cfm-policer2`* statement specifies to police all CFM packets with the specified policer *cfm-policer2*. If the `all` *`policer-name`* option is used, then the user cannot specify the previous `continuity-check` and `other` options.

```
[edit protocols oam ethernet]
connectivity-fault-management {
    policer {
        continuity-check cfm-policer;
```

```
        other cfm-policer1 ;
        all cfm-policer2;
        }
    }
```

Case 2: Session-Level CFM Policing

This example shows a session-level CFM policer used for rate-limiting CFM. The `policer` statement at the session `[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name maintenance-association ma-name]` hierarchy level specifies the policer to use for policing only continuity check packets of the CFM traffic belonging to the specified session. The `other cfm-policer1` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name maintenance-association ma-name]` hierarchy level specifies the policer to use for policing all non-continuity check packets of the CFM traffic belonging to this session only. The `all cfm-policer2` statement specifies to police all CFM packets with the specified policer *cfm-policer2*. If the `all policer-name` option is used, then the user cannot specify the previous `continuity-check` and `other` options.

```
[edit protocols oam ethernet]
connectivity-fault-management {
    maintenance-domain md {
        level number;
        maintenance-association ma {
            continuity-check {
                interval 1s;
            }
            policer {
                continuity-check cfm-policer;
                other cfm-policer1;
                all cfm-policer2;
                }
            }
            mep 1 {
                interface ge-3/3/0.0;
                direction up;
                auto-discovery;
            }
        }
    }
```

In the case of global CFM policing, the same policer is shared across multiple CFM sessions. In per-session CFM policing, a separate policer must be created to rate-limit packets specific to that session.

NOTE: Service-level policer configuration for any two CFM sessions on the same interface at different levels must satisfy the following constraints if the direction of the sessions is the same:

- If one session is configured with `policer all`, then the other session cannot have a `policer all` or `policer other` configuration.
- If one session is configured with `policer other`, then the other session cannot have a `policer all` or `policer other` configuration.

A commit error will occur if such a configuration is committed.

NOTE: Policers with PBB and MIPs are not supported.

SEE ALSO

policer (CFM Session)

policer (CFM Global)

show oam ethernet connectivity-fault-management policer

clear oam ethernet connectivity-fault-management policer

Enabling Enhanced Connectivity Fault Management Mode

You can enable enhanced connectivity fault management (CFM) mode to enable effective Ethernet OAM deployment in scaling networks. On enabling enhanced CFM mode, Junos OS supports 32, 000 maintenance association end points (MEPs) and maintenance intermediate points (MIPs) each per chassis for bridge, VPLS, L2VPN, and CCC domains. In previous releases, Junos OS supports 8, 000 MEPs and 8000 MIPS per chassis. If you do not enable enhanced CFM, Junos OS continues to support existing number of MIPs and MEPs per chassis.

NOTE: To support enhanced CFM mode, configure the network services mode on the router as `enhanced-ip`. If the network services mode is not `enhanced-ip`, and you have enabled enhanced CFM, the following warning message is displayed:

```
[edit protocols oam ethernet] 'connectivity-fault-management' enhanced ip is not effective please configure enhanced ip and give router reboot
```

To enable enhanced CFM mode, perform the following steps:

1. In configuration mode, go to the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management
```

2. Enable effective Ethernet OAM deployment by enabling enhanced CFM mode.

```
[edit protocols oam ethernet connectivity-fault-management ]
user@host# set enhanced-cfm-mode
```

3. Commit the mode change. A warning message is displayed asking you to restart CFM. If you do not restart CFM, CFM is automatically restarted by Junos OS.

```
[edit protocols oam ethernet connectivity-fault-management ]
user@host # commit
[edit protocols oam ethernet]
'connectivity fault management'
CFM mode change is catastrophic. cfmd will be restarted
commit complete
```

4. To verify if the enhanced CFM mode has been configured, use the `show oam ethernet connectivity-fault-management state` command.

```
[edit protocols oam ethernet connectivity-fault-management ]
user@host# show oam ethernet connectivity-fault-management
enhanced-cfm-mode;
traceoptions {
    file cfmd.log size 1g;
}
maintenance-domain md6 {
    level 6;
    maintenance-association ma6 {
        continuity-check {
            interval 1s;
        }
        mep 102 {
            interface ge-0/0/0.0;
            direction up;
        }
```

```
    }
}
```

SEE ALSO

enhanced-cfm-mode

Configure Connectivity Fault Management for Interoperability During Unified In-Service Software Upgrades

Starting in Release 17.1, Junos OS connectivity fault management (CFM), during a unified in-service software upgrade (ISSU), works when the peer device is not a Juniper Networks router. Interoperating with the router of another vendor, the Juniper Networks router retains session information and continues to transmit continuity check message (CCM) PDUs during the unified ISSU. Connectivity fault management continues to operate.

This feature requires the following conditions be met:

- Packet Forwarding Engine keepalives must be enabled to provide inline transmission of CCMs. The feature does not work when the CCMs are transmitted by the CPU of a line card, which is the default transmission method.
- The interval between CCMs must be 1 second.

CFM interoperability during a unified ISSU is supported on the following MPCs: MPC1, MPC2, MPC2-NG, MPC3-NG, MPC5, and MPC6.

To enable CFM interoperability with third-party devices across a unified ISSU:

1. Enable inline keepalives.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# set hardware-assisted-keepalives enable
```

2. Set the CCM interval to 1 second.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name continuity-check]
user@host# set interval 1s
```

SEE ALSO

Enabling Inline Transmission of Continuity Check Messages for Maximum Scaling | 356

Junos OS Support for Performance Monitoring Compliant with Technical Specification MEF 36

Junos OS release 16.1R1 and later supports performance monitoring that is compliant with Technical Specification MEF 36. Technical Specification MEF 36 specifies the performance monitoring MIB. The performance monitoring MIB is required to manage service operations, administration, and maintenance (OAM) implementations that satisfy the Service OAM requirements and framework specified in MEF 17 and MEF 35, the management objects specified in MEF 7.1, and the performance monitoring functions defined in ITU-T Y.1731 and IEEE 802.1ag.

You can enable MEF-36-compliant performance monitoring by configuring the `measurement-interval` statement at the `[edit protocols oam ethernet cfm performance-monitoring]` hierarchy level.

When MEF-36-compliant performance monitoring is enabled:

- An SNMP get next request for a variable might not fetch the current value unless an SNMP walk is performed before performing the get next request. This limitation applies only to the current statistics for delay measurement, loss measurement, and synthetic loss measurement.
- The output for the field `Current delay measurement statistics` might display a measurement interval of 0 (zero) and an incorrect timestamp until the first cycle time has expired.
- Supported data TLV size for performance monitoring protocol data units (PDUs) is 1386 bytes when MEF-36-compliant performance monitoring is enabled. The TLV size is 1400 bytes in legacy mode.
- The maximum configurable value for the lower threshold bin is 4,294,967,294.
- Frame loss ratio (FLR) is excluded in loss measurements during period of unavailability for synthetic loss measurement only. In case of loss measurement, FLR is included even during period of unavailability.
- During a period of loss of continuity (adjacency down), although SOAM PDUs are not sent, FLR and availability calculations are not stopped. These calculations are performed with the assumption of 100% loss.
- The number of SOAM PDUs that are sent during the first measurement interval might be less than expected. This is because of a delay in detecting the adjacency state at the performance monitoring session level.
- The number of SOAM PDUs transmitted during a measurement interval for a cycle time of 100 ms might not be accurate. For example, in a measurement interval of two minutes with a cycle time 100 ms, the SOAM PDUs transmitted might be in the range of 1198—2000.

SEE ALSO

> *measurement-interval*

Damping CFM performance Monitoring Traps and Notifications to Prevent Congestion of The NMS

You can dampen the performance monitoring threshold-crossing traps and notifications that are generated every time a threshold-crossing event occurs to prevent congestion of the network management system (NMS).

Damping limits the number of jnxSoamPmThresholdCrossingAlarm traps sent to the NMS by summarizing the flap occurrences over a period of time, known as the flap trap timer, and sends a single jnxSoamPmThresholdFlapAlarm notification to the NMS. You can configure the duration of the flap trap timer to any value from 1 through 360 seconds.

The jnxSoamPmThresholdFlapAlarm notification is generated and sent when the following conditions are met:

- At least one flap has occurred when the flap timer has expired.
- You changed the value of the flap trap timer, which caused the timer to stop.

You can enable damping at the global level for the iterator or you can enable damping at the individual threshold type of the iterator. For instance, to enable damping at the global level, for the iterator, use the following command: `set protocols oam ethernet cfm performance-monitoring sla-iterator-profiles` *`profile-name`* `flap-trap-monitor`. To enable damping at a specific threshold type, for the `avg-fd-twoway-threshold`, use the following command: `set protocols oam ethernet cfm performance-monitoring sla-iterator-profiles` *`profile-name`* `avg-fdv-twoway-threshold flap-trap-monitor`.

You can also disable damping.

SEE ALSO

> *flap-trap-monitor*
>
> Physical Interface Damping Overview

Release History Table

Release	Description
17.1	Starting in Release 17.1, Junos OS connectivity fault management (CFM), during a unified in-service software upgrade (ISSU), works when the peer device is not a Juniper Networks router.

12.3	Starting with Junos OS Release 12.3, for all interfaces configured on Modular Port Concentrators (MPCs) on MX Series 5G Universal Routing Platforms, you no longer need to configure the `no-control-word` statement for all Layer 2 VPNs and Layer 2 circuits over which you are running CFM MEPs.
12.3	Starting with Junos OS Release 12.3, for all interfaces configured on Modular Port Concentrators (MPCs) on MX Series 5G Universal Routing Platforms, you no longer need to configure the `no-control-word` statement for all Layer 2 VPNs and Layer 2 circuits over which you are running CFM MEPs.
12.3	Starting with Junos OS Release 12.3, for all interfaces configured on Modular Port Concentrators (MPCs) on MX Series 5G Universal Routing Platforms, you no longer need to configure the `no-control-word` statement for all Layer 2 VPNs and Layer 2 circuits over which you are running CFM MEPs.

CFM Action Profile

SUMMARY

IN THIS SECTION

- CFM Action Profile to Bring Down a Group of Logical Interfaces Overview | **55**
- Configure a CFM Action Profile to Bring Down a Group of Logical Interfaces | **57**
- Configure a CFM Action Profile to Specify CFM Actions for CFM Events | **61**

CFM Action Profile to Bring Down a Group of Logical Interfaces Overview

IN THIS SECTION

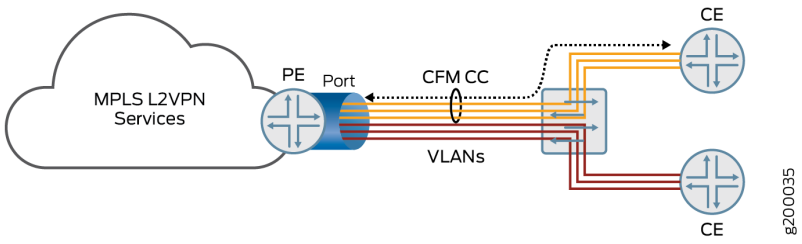
- Benefits of Creating CFM Action Profile to Bring Down a Group of Logical Interfaces | **56**

With growing networks, there is a requirement of monitoring a large number of services using CFM. To monitor each service, one session per service logical interface is required. If the services are large in number, this method does not scale as the number of sessions are limited. Instead of one CFM session per service, a single CFM session can monitor multiple services.

Also, there are scenarios where the user-to-network interface (UNI) device needs to be brought down based on sessions on network-to-network Interface (NNI) logical interface. Here, the NNI logical interface refers to core interface and UNI physical interface refers to access interface hosting multiple service logical interfaces. Based on core interface monitoring, you can bring down service logical interfaces associated with access interface.

Figure 3 on page 56 illustrates a topology where a number of services destined to customer-edge (CE) routers share a single port on a provider-edge (PE) router. Each service uses one logical interface. A set of services or logical interfaces (colored in yellow) are destined to one CE router and a set of services or logical interfaces colored in red are destined to another CE router. To monitor each service, you need dedicated down maintenance association end point (MEP) sessions for each service. You can bring down the service by bringing down the service logical interface whenever the session goes down. However, this approach is not scalable if we have large number of services. Monitoring the CFM session on the physical interface is also not feasible because multiple CE routers might be connected and the services to other CE router could be disrupted. To address this issue of monitoring multiple services with a single session, you can create a CCM action profile to bring down a group of logical interfaces by using a CFM session that is configured on a single logical interface.

Figure 3: Topology of Multiple VLAN Services Sharing a Single Port on PE Router Destined to Multiple CE Routers



You can configure CCM action profiles for the following scenarios:

- To bring down a group of logical interfaces all having the same parent port when CCM monitoring session is running on one of the logical interface but on a different parent port.
- To bring down a group of logical interfaces when CCM monitoring session is running on one of the logical interfaces, all belonging to the same parent port.
- To bring down the port, when the CCM monitoring session is running on one of the logical interfaces of a different parent port.

Benefits of Creating CFM Action Profile to Bring Down a Group of Logical Interfaces

- Reduces resource requirement in scaled networks where multiple services need to be monitored.

- Avoids the need to create individual MEP sessions for each service in a topology that includes multiple services to be monitored, thereby enhancing the performance and scalability of the network.

SEE ALSO

action-profile

Configure a CFM Action Profile to Bring Down a Group of Logical Interfaces

To monitor multiple services or IFLs using CFM session configured on a single logical interface, you can create a CCM action profile to bring down a group of logical interfaces. You need to define an action to bring down the interface group in the action profile. You will then define the interface device name and the number of logical interfaces that have to be brought down. A logical interface is represented by a combination of the `interface-device-name` and `unit-list`. The following steps explain the procedure to bring down a group of logical interfaces when the `interface-device-name` and/or `unit-list` are specified.

1. In configuration mode, at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level, specify the name of the action profile and the CFM event(s). You can configure more than one event in the action profile.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set action-profile profile-name event [event1, event2, event3..]
```

For example,

```
user@host# set action-profile AP_test event adjacency-loss rdi
```

NOTE: The action `interface-group-down` will not be supported with events other than adjacency-loss and RDI. Any other events configured results in a commit error.

2. In configuration mode, at the [`edit protocols oam ethernet connectivity-fault-management action-profile profile-name`] hierarchy level, define the action to bring down the interface group.

```
[edit protocols oam ethernet connectivity-fault-management action-profile  AP-test ]
user@host# set action interface-group-down
```

NOTE: The action `interface-group-down` will not be supported with other interface related actions. Any other actions configured results in a commit error.

3. At the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level, define the maintenance domain. Specify the maintenance-association parameters.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenance-domain domain-name  level number maintenance-association ma-name
continuity-check interval 1s
```

For example,

```
user@host# set maintenance-domain md6 level 6 maintenance-association ma6 continuity-check
interval 1s
```

4. At the `edit protocols oam ethernet connectivity-fault-management maintenance-domain` *md-name* `maintenance-association` *ma-name*, define the maintenance association endpoint and the associated parameters.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name]
user@host# set mep mep-id interface interface-name direction down remote -mep mep-id
```

For example,

```
user@host# set mep 101 interface ge-0/0/0.0 direction down remote -mep 102
```

5. If the action-profile has `interface-group-down` action configured, it is mandatory to configure the `interface-group` at the RMEP level. In the configuration mode at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *md-name* `maintenance-association` *ma-name* `mep` *mep-id* `remote-mep` *mep-id* `action-profile` *profile-name*] include the `interface-group` statement to bring down the interface group marked with the action profile as `interface-group-down`.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep mep-id action-profile profile-name]
user@host# set interface-group
```

For example,

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md6 maintenance-
association ma6 mep 101 remote-mep 102 action-profile AP_test]
user@host# set interface-group
```

NOTE: If the `interface-group` configuration is not included in the RMEP configuration. The configuration results in commit error.

6. A logical interface is represented by a combination of the `interface-device-name` and `unit-list`. Configure the device interface name and the number of logical interfaces at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* `mep` *`mep-id`* `remote-mep` *`mep-id`* `action-profile` *`profile-name`* `interface-group`.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep mep-id action-profile profile-name
interface-group]
user@host# set interface interface-name
user@host# set unit-list logical-interface-unit-number
```

For example,

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md6 maintenance-
association ma6 mep 101 remote-mep 102 action-profile AP_test interface-group]
user@host# set interface ge-0/0/0.0
user@host# set unit-list 1223-3344
```

In this configuration example, the interface ge-0/0/0.0 is brought down.

NOTE:

- At least one of the `interface-group` parameters, `interface-device-name` or `unit-list` must be configured. If the interface device name is not configured, the MEP interface is considered as the device name and the logical interface on that device is brought down.
- If the `unit-list` parameter exceeds the recommended limit, a commit error occurs.

- If the *interface-device-name* is not specified in the `interface-group`, the logical interface numbers mentioned in `unit-list` for the physical interface is brought down.
- If the *unit-list* is not specified in the `interface-group`, IFLs are brought down for the configured interface.

7. Verify the configuration using `show protocols oam` command.

```
[edit]
user@host# show protocols oam
ethernet {
    connectivity-fault-management {
        action-profile AP_TEST {
            event {
                adjacency-loss;
                rdi;
            }
            action {
                interface-group-down;
            }
        }
        maintenance-domain md6 {
            level 6;
            maintenance-association ma6 {
                continuity-check {
                    interval 1s;
                }
                mep 102 {
                    interface ge-0/0/0.0;
                    direction down;
                    remote-mep 103 {
                        action-profile AP_TEST;
                        interface-group {
                          ge-0/0/1;
                           unit-list [12 23-33 44];
                        }
                    }
                }
            }
        }
```

```
    }
}
```

SEE ALSO

interface-group

interface-group-down

Configure a CFM Action Profile to Specify CFM Actions for CFM Events

You can create a connectivity fault management (CFM) action profile to define event flags and thresholds to be monitored. You can also specify the action to be taken when any of the configured events occur. When the CFM events occur, the router performs the corresponding action based on your specification. You can configure one or more events in the action profile. Alternatively, you can configure an action profile and specify default actions when connectivity to a remote maintenance association endpoint (MEP) fails.

NOTE: You cannot configure multiple actions at this time. Only one action can be configured. This limitation affects both the `action` and `clear-action` statements.

To configure the CFM action profile:

1. In configuration mode, at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level, specify the name of the action profile and the CFM event(s). You can configure more than one event in the action profile. Possible events include: interface-status-tlv, port-status-tlv, adjacency-loss, RDI.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set action-profile profile-name event [event1, event2, event3..]
```

2. Specify the action to be taken by the router when the event occurs. The action is triggered when the event occurs. If you have configured more than one event in the action profile, it is not necessary for all events to occur to trigger the action.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set action-profile profile-name action action
```

3. Specify the default action to be taken by the router when connectivity to a remote MEP fails. If no action is configured, no action is taken.

NOTE: Associating an action profile with the `interface-down` action on an up MEP CFM session running over a circuit cross-connect (CCC) interface (l2circuit/l2vpn) is not advisable and can result in a deadlock situation.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set action-profile profile-name default-actions action
```

SEE ALSO

event (CFM)

default-actions

connectivity-fault-management

Ethernet Local Management Interface

IN THIS SECTION

- Ethernet Local Management Interface Overview | 62
- Configure the Ethernet Local Management Interface | 65
- Example E-LMI Configuration | 67

Ethernet Local Management Interface Overview

Gigabit Ethernet (ge), 10-Gigabit Ethernet (xe), and Aggregated Ethernet (ae) interfaces support the Ethernet Local Management Interface (E-LMI).

NOTE: On MX Series routers, E-LMI is supported on Gigabit Ethernet (ge), 10-Gigabit Ethernet (xe), and Aggregated Ethernet (ae) interfaces configured on MX Series routers with DPC only.

The E-LMI specification is available at the Metro Ethernet Forum. E-LMI procedures and protocols are used for enabling automatic configuration of the customer edge (CE) to support Metro Ethernet services. The E-LMI protocol also provides user-to-network interface (UNI) and Ethernet virtual connection (EVC) status information to the CE. The UNI and EVC information enables automatic configuration of CE operation based on the Metro Ethernet configuration.

The E-LMI protocol operates between the CE device and the provider edge (PE) device. It runs only on the PE-CE link and notifies the CE of connectivity status and configuration parameters of Ethernet services available on the CE port. The scope of the E-LMI protocol is shown in Figure 4 on page 63.

Figure 4: Scope of the E-LMI Protocol

The E-LMI implementation on ACX and MX Series routers includes only the PE side of the E-LMI protocol.

E-LMI interoperates with an OAM protocol, such as Connectivity Fault Management (CFM), that runs within the provider network to collect OAM status. CFM runs at the provider maintenance level (UNI-N to UNI-N with up MEPs at the UNI). E-LMI relies on the CFM for end-to-end status of EVCs across CFM domains (SVLAN domain or VPLS).

The E-LMI protocol relays the following information:

- Notification to the CE of the addition/deletion of an EVC (active, not active, or partially active)
- Notification to the CE of the availability state of a configured EVC
- Communication of UNI and EVC attributes to the CE:
  - UNI attributes:
    - UNI identifier (a user-configured name for UNI)
    - CE-VLAN ID/EVC map type (all-to-one bundling, service multiplexing with bundling, or no bundling)
    - Bandwidth profile is not supported (including the following features):

        - CM (coupling mode)
        - CF (color flag)
        - CIR (committed Information rate)
        - CBR (committed burst size)
        - EIR (excess information rate)
        - EBS (excess burst size)
    - EVC attributes:
        - EVC reference ID
        - EVC status type (active, not active, or partially active)
        - EVC type (point-to-point or multipoint-to-multipoint)
        - EVC ID (a user-configured name for EVC)
        - Bandwidth profile (not supported)
    - CE-VLAN ID/EVC map

E-LMI on MX Series routers supports the following EVC types:

- Q-in-Q SVLAN (point-to-point or multipoint-to-multipoint)—Requires an end-to-end CFM session between UNI-Ns to monitor the EVS status.
- VPLS (BGP or LDP) (point-to-point or multipoint-to-multipoint)—Either VPLS pseudowire status or end-to-end CFM sessions between UNI-Ns can be used to monitor EVC status.
- L2 circuit/L2VPN (point-to-point)—Either VPLS pseudowire status or end-to-end CFM sessions between UNI-Ns can be used to monitor EVC status.

> **NOTE**: `l2-circuit` and `l2vpn` are not supported.

The E-LMI protocol on ACX Series routers supports Layer 2 circuit and Layer 2 VPN EVC types and enables link-loss forwarding for pseudowire (Layer 2 circuit and Layer 2 VPN) services as follows:

- Interworking between the connectivity fault management (CFM) protocol and the E-LMI protocol for Layer 2 circuit and Layer 2 VPN.
    - End-to-end CFM session between UNIs to monitor EVC status.

  - In the case of pseudowire redundancy, CFM can be used to monitor active and backup pseudowire sessions. The EVC status is declared as down to CE devices only when both the active and backup pseudowire sessions go down.

- Interworking between remote defect indication (RDI) and E-LMI for Layer 2 circuit and Layer 2 VPN.

  - If a maintenance association end point (MEP) receives an RDI bit set in a continuity check message (CCM) frame, and if RDI fault detection is enabled in the EVC configuration at `[edit protocols oam ethernet evcs evc-id evc-protocol cfm management-domain name management-association name faults rdi]`, then the pseudowire is declared as down to CE routers through E-LMI.

- If an end-to-end CFM session does not exist between UNIs, the pseudowire (Layer 2 circuit or Layer 2 VPN) up and down state triggers an asynchronous EVC state change message to CE routers through E-LMI.

NOTE: ACX Series routers do not support E-LMI for Layer 2 services (bridging).

Configure the Ethernet Local Management Interface

IN THIS SECTION

- Configuring an OAM Protocol (CFM) | **65**
- Assigning the OAM Protocol to an EVC | **65**
- Enabling E-LMI on an Interface and Mapping CE VLAN IDs to an EVC | **66**

To configure E-LMI, perform the following steps:

Configuring an OAM Protocol (CFM)

For information on configuring the OAM protocol (CFM), see "IEEE 802.1ag OAM Connectivity Fault Management Overview" on page 21.

Assigning the OAM Protocol to an EVC

To configure an EVC, you must specify a name for the EVC using the `evcs`*`evc-id`* statement at the `[edit protocols oam ethernet]` hierarchy level. You can set the EVC protocol for monitoring EVC statistics to `cfm` or `vpls` using the `evc-protocol` statement and its options at the `[edit protocols oam ethernet evcs]` hierarchy level.

You can set the number of remote UNIs in the EVC using the `remote-uni-count` *number* statement at the `[edit protocols oam ethernet evcs evcs-protocol]` hierarchy level. The `remote-uni-count` defaults to 1. Configuring a value greater than 1 makes the EVC multipoint-to-multipoint. If you enter a value greater than the actual number of endpoints, the EVC status will display as partially active even if all endpoints are up. If you enter a `remote-uni-count` less than the actual number of endpoints, the status will display as active, even if all endpoints are not up.

You can configure an EVC by including the `evcs` statement at the `[edit protocols oam ethernet]` hierarchy level:

```
[edit protocols oam ethernet]
evcs evc-id {
    evc-protocol (cfm (management-domain name management-association name ) | vpls (routing-
instance name)) {
        remote-uni-count <number>;      # Optional, defaults to 1
        multipoint-to-multipoint;
 # Optional, defaults to point-to-point if remote-uni-count is 1
    }
}
```

Enabling E-LMI on an Interface and Mapping CE VLAN IDs to an EVC

To configure E-LMI, include the `lmi` statement at the `[edit protocols oam ethernet]` hierarchy level:

```
[edit protocols oam ethernet]
lmi {
    polling-verification-timer value;
# Polling verification timer (T392), defaults to 15 seconds
    status-counter count; # Status counter (N393), defaults to 4
    interface name {
        evc evc-id {
            default-evc;
            vlan-list [ vlan-ids ];
        }
        evc-map-type (all-to-one-bundling | bundling | service-multiplexing);
        polling-verification-time value; # Optional, defaults to global value
        status-counter count; # Optional, defaults to global value
        uni-id value; # Optional, defaults to interface-name
    }
}
```

You can set the status counter to count consecutive errors using the `status-counter` *`count`* statement at the `[edit protocols oam ethernet lmi]` hierarchy level. The status counter is used to determine if E-LMI is operational or not. The default value is 4.

You can set the `polling-verification-timer` *`value`* statement at the `[edit protocols oam ethernet lmi]` hierarchy level. The default value is 15 seconds.

You can enable an interface and set its options for use with E-LMI using the `interface` *`name`* statement at the `[edit protocols oam ethernet lmi]` hierarchy level. Only `ge`, `xe`, and `ae` interfaces are supported. You can use the interface `uni-id` option to specify a name for the UNI. If `uni-id` is not configured, it defaults to the name variable of `interface` *`name`*.

You can specify the CE-VLAN ID/EVC map type using the `evc-map-type type` interface option. The options are `all-to-one-bundling`, `bundling`, or `service-multiplexing`. Service multiplexing is with no bundling. The default type is `all-to-one-bundling`.

To specify the EVC that an interface uses, use the `evc` *`evc-id`* statement at the `[edit protocols oam ethernet lmi interface` *`name`*`]` hierarchy level. You can specify an interface as the default EVC interface using the `default-evc` statement at the `[edit protocols oam ethernet lmi interface` *`name`* `evc` *`evc-id`*`]` hierarchy level. All VIDs that are not mapped to any other EVCs are mapped to this EVC. Only one EVC can be configured as the default.

You can map a list of VLANs to an EVC using the `vlan-list` *`vlan-id-list`* statement at the `[edit protocols oam ethernet lmi interface` *`name`* `evc` *`evc-id`*`]` hierarchy level.

Example E-LMI Configuration

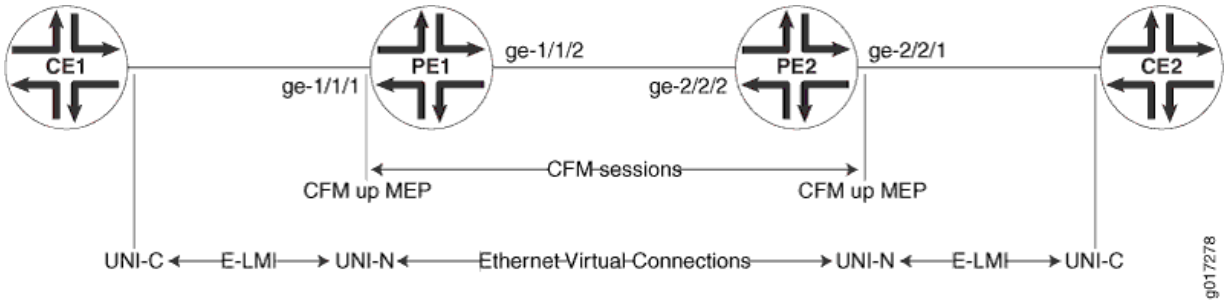
IN THIS SECTION

- Example Topology | **67**
- Configuring PE1 | **68**
- Configuring PE2 | **70**
- Configuring Two UNIs Sharing the Same EVC | **72**

Example Topology

Figure 5 on page 68 illustrates the E-LMI configuration for a point-to-point EVC (SVLAN) monitored by CFM. In this example, VLANs 1 through 2048 are mapped to `evc1` (SVLAN 100) and 2049 through 4096 are mapped to `evc2` (SVLAN 200). Two CFM sessions are created to monitor these EVCs.

Figure 5: E-LMI Configuration for a Point-to-Point EVC (SVLAN) Monitored by CFM



Configuring PE1

```
[edit]
interfaces {
    ge-1/1/1 {
        unit 0 {
            family bridge {
                interface-mode trunk;
                vlan-id-list 1-2048;
            }
        }
        unit 1 {
            family bridge {
                interface-mode trunk;
                vlan-id-list 2049-4096;
            }
        }
    }
    ge-1/1/2 {
        unit 0 {
            vlan-id 100;
            family bridge {
                interface-mode trunk;
                inner-vlan-id-list 1-2048;
            }
        }
        unit 1 {
            vlan-id 200;
            family bridge {
                interface-mode trunk;
                inner-vlan-id-list 2049-4096;
```

```
            }
        }
    }
}
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain md {
                    level 0;
                    maintenance-association 1 {
                        name-format vlan;
                        mep 1 {
                            direction up;
                            interface ge-1/1/1.0 vlan 1;
                        }
                    }
                    maintenance-association 2049 {
                        name-format vlan;
                        mep 1 {
                            direction up;
                            interface ge-1/1/1.1 vlan 2049;
                        }
                    }
                }
            }
            evcs {
                evc1 {
                    evc-protocol cfm management-domain md management-association 1;
                    remote-uni-count 1;
                }
                evc2 {
                    evc-protocol cfm management-domain md management-association 2049;
                    remote-uni-count 1;
                }
            }
            lmi {
                interface ge-1/1/1 {
                    evc evc1 {
                        vlan-list 1-2048;
                    }
                    evc evc2 {
                        vlan-list 2049-4096;
```

```
                    }
                    evc-map-type bundling;
                    uni-id uni-ce1;
                }
            }
        }
    }
}
```

Configuring PE2

```
[edit]
interfaces {
    ge-2/2/1 {
        unit 0 {
            family bridge {
                interface-mode trunk;
                vlan-id-list 1-2048;
            }
        }
        unit 1 {
            family bridge {
                interface-mode trunk;
                vlan-id-list 2049-4096;
            }
        }
    }
    ge-2/2/2 {
        unit 0 {
            vlan-id 100;
            family bridge {
                interface-mode trunk;
                inner-vlan-id-list 1-2048;
            }
        }
        unit 1 {
            vlan-id 200;
            family bridge {
                interface-mode trunk;
                inner-vlan-id-list 2049-4095;
            }
```

```
        }
    }
}
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain md {
                    level 0;
                    maintenance-association 1 {
                        name-format vlan;
                        mep 1 {
                            direction up;
                            interface ge-2/2/1.0 vlan 1;
                        }
                    }
                    maintenance-association 2049 {
                        name-format vlan;
                        mep 1 {
                            direction up;
                            interface ge-2/2/1.1 vlan 2049;
                        }
                    }
                }
            }
            evcs {
                evc1 {
                    evc-protocol cfm management-domain md management-association 1;
                    remote-uni-count 1;
                }
                evc2 {
                    evc-protocol cfm management-domain md management-association 2049;
                    uni-count 2;
                }
            }
            lmi {
                interface ge-2/2/1 {
                    evc evc1 {
                        vlan-list 1-2048;
                    }
                    evc evc2 {
                        vlan-list 2049-4095;
                    }
```

```
                    evc-map-type bundling;
                    uni-id uni-ce2;
                }
            }
        }
    }
}
```

Configuring Two UNIs Sharing the Same EVC

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management { ...}
        evcs {
            evc1 {
                evc-protocol cfm management-domain md management-association 1;
                remote-uni-count 1;
            }
        }
        lmi {
            interface ge-2/2/1 {
                evc evc1 {
                    vlan-list 0-4095;
                }
                evc-map-type all-to-one-bundling;
                uni-id uni-ce1;
            }
            interface ge-2/3/1 {
                evc evc1 {
                    vlan-list 0-4095;
                }
                evc-map-type all-to-one-bundling;
                uni-id uni-ce2;
            }
        }
    }
}
```

RELATED DOCUMENTATION

connectivity-fault-management

CFM Support for CCC Encapsulated Packets

IN THIS SECTION

- IEEE 802.1ag CFM OAM Support for CCC Encapsulated Packets Overview | 73
- CFM Features Supported on Layer 2 VPN Circuits | 73
- Configure CFM for CCC Encapsulated Packets | 74

IEEE 802.1ag CFM OAM Support for CCC Encapsulated Packets Overview

Layer 2 virtual private network (L2VPN) is a type of virtual private network service used to transport customer's private Layer 2 traffic (for example, Ethernet, ATM or Frame Relay) over the service provider's shared IP/MPLS infrastructure. The service provider edge (PE) router must have an interface with circuit cross-connect (CCC) encapsulation to switch the customer edge (CE) traffic to the public network.

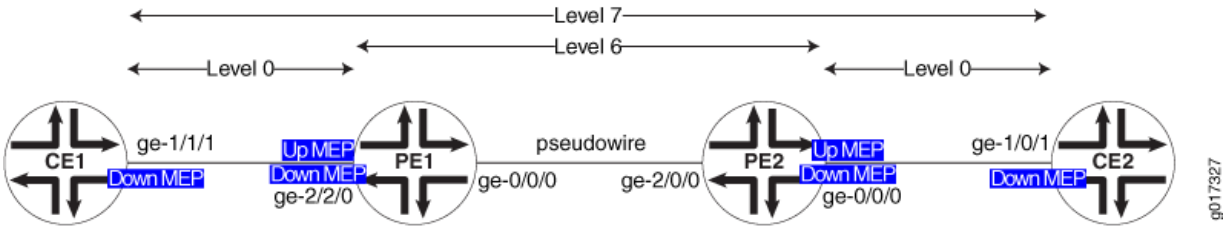
The IEEE 802.1ag Ethernet Connectivity Fault Management (CFM) is an OAM standard used to perform fault detection, isolation, and verification on virtual bridge LANs. M120 and MX Series routers provide CFM support for bridge/VPLS/routed interfaces and support 802.1ag Ethernet OAM for CCC encapsulated packets.

CFM Features Supported on Layer 2 VPN Circuits

CFM features supported on L2VPN circuits are as follows:

- Creation of up/down MEPs at any level on the CE-facing logical interfaces.
- Creation of MIPs at any level on the CE-facing logical interfaces.
- Support for continuity check, loopback, and linkrace protocol.
- Support for the Y1731 Ethernet Delay measurement protocol.
- Support for action profiles to bring the CE-facing logical interfaces down when loss of connectivity is detected.

Figure 6: Layer 2 VPN Topology



To monitor the L2VPN circuit, a CFM up MEP (Level 6 in Figure 6 on page 74) can be configured on the CE-facing logical interfaces of provider edge routers PE1 and PE2. To monitor the CE-PE attachment circuit, a CFM down MEP can be configured on the customer logical interfaces of CE1-PE1 and CE2-PE2 (Level 0 in Figure 6 on page 74).

Configure CFM for CCC Encapsulated Packets

The only change from the existing CLI configuration is the introduction of a new command to create a MIP on the CE-facing interface of the PE router.

```
protocols {
    oam {
        ethernet {
            connectivity-fault-management {

                # Define a maintenance domains for each default level.
                #; These names are specified as DEFAULT_level_number
                maintenance-domain DEFAULT_x {
                    # L2VPN CE interface
                    interface (ge | xe)-fpc/pic/port.domain;
                }
                {
                level number;
                maintenance-association identifier {
                    mep mep-id {
                        direction (up | down);
                        # L2 VPN CE interface on which encapsulation family CCC is configured.
                        interface  (ge | xe)-fpc/pic/port.domain;
                        auto-discovery;
                        priority number;
                    }
                }
            }
        }
```

```
    }
    }
}
```

SEE ALSO

connectivity-fault-management

Configure Unified ISSU for 802.1ag CFM

A unified in-service software upgrade (ISSU) enables you to upgrade between two different Junos OS releases with no disruption on the control plane and with minimal disruption of traffic. Unified ISSU is automatically enabled for the Connectivity Fault Management (CFM) protocols and interoperates between local and remote maintenance endpoints (MEPs).

The Junos OS provides support for unified ISSU using the loss threshold type length value (TLV), which is automatically enabled for CFM. TLVs are described in the IEEE 802.1ag standard for CFM as a method of encoding variable-length and optional information in a protocol data unit (PDU). The loss threshold TLV indicates the loss threshold value of a remote MEP. The loss threshold TLV is transmitted as part of the CFM continuity check messages.

NOTE: Starting in Junos OS Release 15.1, configuring ISSU with CFM (802.1ag) is supported only on MX and PTX routers that support TLV. Interoperation with other vendors is not supported.

During a unified ISSU, the control plane may go down for several seconds and cause CFM continuity check packets to get dropped. This may cause the remote MEP to detect a connectivity loss and mark the MEP as down. To keep the MEP active during a unified ISSU, the loss threshold TLV communicates the minimum threshold value the receiving MEP requires to keep the MEP active. The receiving MEP parses the TLV and updates the loss threshold value, but only if the new threshold value is greater than the locally configured threshold value.

An overview of CFM is described starting in "IEEE 802.1ag OAM Connectivity Fault Management Overview" on page 21, and you should further observe the additional requirements described in this topic.

Table 6 on page 76 shows the Loss Threshold TLV format.

Table 6: Loss Threshold TLV Format

Parameter	Octet (sequence)	Description
Type=31	1	Required. Required. If 0, no Length or Value fields follow. If not 0, at least the Length field follows the Type field.
Length=12	2	Required if the Type field is not 0. Not present if the Type field is 0. The 16 bits of the Length field indicate the size, in octets, of the Value field. 0 in the Length field indicates that there is no Value field.
OUI	3	Optional. Organization unique identifier (OUI), which is controlled by the IEEE and is typically the first three bytes of a MAC address (Juniper OUI 0x009069).
Subtype	1	Optional. Organizationally defined subtype.
Value	4	Optional. Loss threshold value.
Flag	4	Optional. Bit0 (identifies an ISSU is in progress) Bit1-31 (reserved)

Junos OS provides configuration support for the `convey-loss-threshold` statement, allowing you to control the transmission of the loss threshold TLV in continuity check messages PDUs. The `convey-loss-threshold` statement specifies that the loss threshold TLV must be transmitted as part of the continuity check messages. If the `convey-loss-threshold` statement is not specified, continuity check messages transmit this TLV only when a unified ISSU is in progress. The Junos OS provides this configuration at the continuity-check level. By default, continuity check messages do not include the loss threshold TLV.

To configure the convey loss threshold, use the `convey-loss-threshold` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`identifier`* `maintenance-association` *`identifier`* `continuity-check]` hierarchy level.

For the remote MEP, the loss threshold TLV is transmitted only during the unified ISSU if the `convey-loss-threshold` statement is not configured. The remote MEP switches back to the default loss threshold if no loss threshold TLV is received or the TLV has a default threshold value of 3.

An example of the ISSU configuration statements follows:

```
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain identifier {
                level number;
                    maintenance-association identifier {
                        continuity-check {
                            convey-loss-threshold;
                            interval number;
                            loss-threshold number;
                            hold-interval number;
                        }
                    }
                }
            }
        }
    }
}
```

The Junos OS saves the last received loss threshold TLV from the remote MEP. You can display the last saved loss threshold TLV that is received by the remote MEP, using the `show oam ethernet connectivity-fault-management mep-database maintenance-domain` *identifier* `maintenance-association` *identifier* `local-mep` *identifier* `remote-mep` *identifier* command, as in the following example:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md3
maintenance-association ma5 local-mep 2 remote-mep 1
Maintenance domain name: md3, Format: string, Level: 3
  Maintenance association name: ma3, Format: string
  Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
  MEP identifier: 2, Direction: up, MAC address: 00:19:e2:b0:76:be
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: none, Port status TLV: none
  Connection Protection TLV: yes
    Prefer me: no, Protection in use: no, FRR Flag: no
  Interface name: xe-4/1/1.0, Interface status: Active, Link status: Up
    Loss Threshold TLV:
      Loss Threshold: 3  , Flag:    0x0
```

```
 Remote MEP identifier: 1, State: ok
    MAC address: 00:1f:12:b7:ce:79, Type: Learned
    Interface: xe-4/1/1.0
    Last flapped: Never
    Continuity: 100%, Admin-enable duration: 45sec, Oper-down duration: 0sec
    Effective loss threshold: 3 frames
    Remote defect indication: false
    Port status TLV: none
    Interface status TLV: none
    Connection Protection TLV:
      Prefer me: no, Protection in use: no, FRR Flag: no
    Loss Threshold TLV:   #Displays last received value
      Loss Threshold: 3  , Flag:    0x0
```

The Junos OS saves the last transmitted loss threshold TLV from a local MEP. You can display the last transmitted loss threshold TLV and the effective loss (operational) threshold for the remote MEP, using the `show oam ethernet connectivity-fault-management mep-database maintenance-domain` *identifier* `maintenance-association` *identifier* `local-mep` *identifier* `remote-mep` *identifier* command, as in the following example:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md3
maintenance-association ma5 local-mep 2 remote-mep 1
Maintenance domain name: md3, Format: string, Level: 3
  Maintenance association name: ma3, Format: string
  Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
  MEP identifier: 2, Direction: up, MAC address: 00:19:e2:b0:76:be
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: none, Port status TLV: none
  Connection Protection TLV: yes
    Prefer me: no, Protection in use: no, FRR Flag: no
  Interface name: xe-4/1/1.0, Interface status: Active, Link status: Up
    Loss Threshold TLV:   #Displays last transmitted value
      Loss Threshold: 3  , Flag:    0x0

   Remote MEP identifier: 1, State: ok
    MAC address: 00:1f:12:b7:ce:79, Type: Learned
    Interface: xe-4/1/1.0
    Last flapped: Never
    Continuity: 100%, Admin-enable duration: 45sec, Oper-down duration: 0sec
    Effective loss threshold: 3 frames   #Displays operational threshold
    Remote defect indication: falsePort status TLV: none
    Interface status TLV: none
```

```
    Connection Protection TLV:
      Prefer me: no, Protection in use: no, FRR Flag: no
    Loss Threshold TLV:
      Loss Threshold: 3   , Flag:    0x0
```

Release History Table

Release	Description
15.1	Starting in Junos OS Release 15.1, configuring ISSU with CFM (802.1ag) is supported only on MX and PTX routers that support TLV.

RELATED DOCUMENTATION

- Before You Begin a Unified ISSU
- Unified ISSU System Requirements

CFM Monitoring between CE and PE Devices

IN THIS SECTION

- CFM Action Profile Asynchronous Notification | 80
- Configuring a CFM Action Profile to Asyncronus Notification | 80
- Understanding CFM Monitoring between CE and PE Devices | 83
- Configuring Port Status TLV and Interface Status TLV | 85
- Configuring Chassis ID TLV | 100
- Configuring MAC Flush Message Processing in CET Mode | 101
- Example: Configuring an Action Profile Based on Connection Protection TLVs | 104

Use this topic to understand more about CFM monitoring between provider edge devices and customer edge devices when the customer edge device is not a Juniper device. Also, you can understand more about how Interface Status TLVs, port status TLVs, chassis ID TLV, and connection protection TLV help in monitoring your network.

CFM Action Profile Asynchronous Notification

SUMMARY

CFM driven asynchronous notification enables link status synchronization between two CE devices connectedto each other through a pseudo wire originating from their respective PE devices It emulatesthe scenario as if two CE devicesare directly connected. CFM provides end-to-end signaling even if PE1 and PE2 are not connected through single network but a set of networks.

Layer 2 connectivity between PE1 and PE2

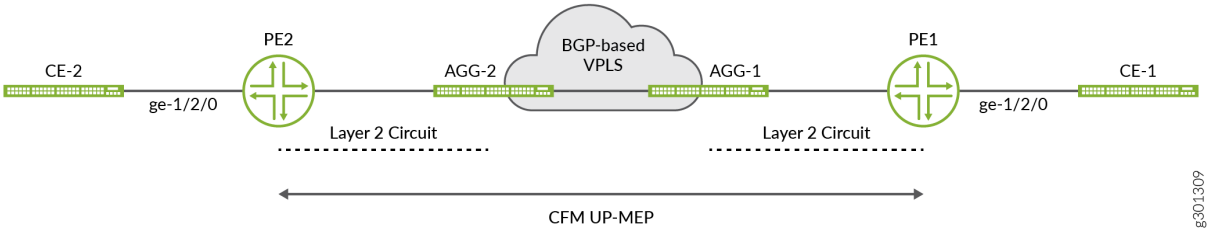


Figure 1 is an example of deployment scenario where CFM based asynchronous-notification can be used to synchronize link status between CE1 and CE2. Following two requirements can be met with the configuration of asynchronous-notification.

- When the link between PE2 to CE2 goes down then the link between PE1 to CE1 also made down.When the link is restored, it should restore the link status PE1 to CE1 also. The link status change between PE1 to CE1 should work the similar way.
- When there is a connectivity issue between PE1 to PE2, its triggers a link down between PE1 to CE1 and PE2 to CE2. If the connection status is restored, it should restore the link status on both ends

SEE ALSO

connectivity-fault-management

Configuring a CFM Action Profile to Asyncronus Notification

SUMMARY

CFM UP-MEP on PE1 to PE2, monitors the connectivity between PE1 to PE2. Use of `interface-status-tlv` on these UP-MEP end points conveys the link status between PE1 to CE1 to PE2 and link-status between PE2 to CE2 to PE1. Action profile must be configured on PE1 to PE2 to drive asynchronous-notification towards respective CE devices . It is triggered when either adjacency-loss is detected or link-down is detected in the received `interface-status-tlv`.

1. Enable `asynchronous-notification` at interface level

   For example

   ```
   user@host# set interface interface-name gigether-option asynchronous-notification
   ```

2. Configure the action profile and the CFM event(s) to triggered this action profile at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level. You can configure more than one event in the action profile

   ```
   [edit protocols oam ethernet connectivity-fault-management]
   user@host# set action-profile profile-name event [event1, event2, event3]
   ```

   For example

   ```
   user@host# set action-profile AP_test event adjacency-loss
   ```

   The action `asynchronous-notification` is not supported with events other than `interface-status-tlv down`, `interface-status-tlv lower-layer-down` and `adjacency-loss`. Any other events configured results in a commit error

   .

3. Define the action to asynchronous-notification at the [`edit protocols oam ethernet connectivity-fault-management` action-profile profile-name] hierarchy level.

   ```
   [edit protocols oam ethernet connectivity-fault-management action-profile AP_test]
   user@host# set action asynchronous-notification
   ```

4. Define the maintenance domain at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level and specify the maintenance-association parameters

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# set maintenance-domain domain-name level number maintenance-association ma-name
continuity-check interval 1s
```

For example

```
user@host# set maintenance-domain md6 level 6 maintenance-association ma6 continuity-check
interval 1s
```

5. Configure the generation of `interface-status-tlv` .it is required if `asynchronous-notification` configured based on `interface-status-tlv`.

```
[edit protocols oam ethernet connectivity-fault-management] user@host# set maintenance-domain
domain-name level number maintenance-association ma-name
continuity-check interface-status-tlv
```

For example

```
user@host# set maintenance-domain md6 level 6 maintenance-association ma6 continuity-check
interface-status-tlv
```

6. Define the maintenance association endpoint at the [`edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`*] hierarchy level and specify the associated parameters.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name] user@host# set mep mep-id direction up interface interface-
name
```

For example

```
user@host# set mep 101 direction up interface ge-0/0/0.0
```

7. Set asynchronous-notification action profile at the RMEP level.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep mep-id ] user@host# set action profile
action profile-name
```

For example,

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md6 maintenance-
association ma6 mep 101 remote-mep 102] user@host# set action-profile AP_test
```

SEE ALSO

No Link Title

Understanding CFM Monitoring between CE and PE Devices

IN THIS SECTION

- Single Active Multi-homing Use Case using RDI bit | 84
- Active/Active Multihoming Use case using RDI bit | 84

You can enable connectivity fault management (CFM) monitoring between provider edge devices and customer edge devices when the customer edge device is not a Juniper device. When the interface is down, CFM propagates the status of the interface in the CC messages. The CC message informs the customer edge device that the provider edge device is down.

You can configure CFM monitoring using either of the following two options:

- Interface Status TLV (Type, Length, and Value)—You can enable connectivity fault management (CFM) monitoring between provider edge devices and customer edge devices when the customer edge device is not a Juniper device by using Interface Status TLV. When the interface is down, CFM propagates the status of the interface using interface status TLV. The Interface Status TLV indicates the status of the interface on which the MEP transmitting the CCM is configured, or the next-lower interface in the IETF RFC 2863 IF-MIB. Thus, the customer edge device is aware that the provider edge device is down. To configure CFM monitoring using Interface Status TLV, use the `interface-status-tlv` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain`

*maintenance-domain*maintenance-association *maintenance-association* continuity-check hierarchy level. This is the standard option.

- RDI (Remote Defect Indication)—Starting in Junos OS Release 17.3R1, you can enable connectivity fault management (CFM) monitoring between provider edge devices and customer edge devices when the customer edge device is not a Juniper device by using the remote defect indication (RDI) bit. When you enable CFM monitoring, CFM propagates the status of the provider edge device via the remote defect indication (RDI) bit in the CC messages. Thus, the customer edge device is aware that the provider edge device is down. The RDI bit is cleared when the service is back up. To configure CFM monitoring using the RDI bit, use the interface-status-send-rdi statement at the [edit protocols oam ethernet connectivity-fault-management maintenance-domain *maintenance-domain*maintenance-association *maintenance-association* continuity-check hierarchy level. This option is required if the customer edge device does not support Interface Status TLV.

NOTE: When the interface is set to CCC down and you have configured RDI, then RDI bit is sent. CFM does not monitor the status of the interface. If CCC down is set when the interface is not standby, RDI bit is sent with the CC messages if you have configured RDI.

Single Active Multi-homing Use Case using RDI bit

Consider the following topology where there are two provider edge devices (PE1 and PE2) as well as two customer edge devices (CE1 and CE2). PE1 is in active state while PE2 is in standby state. CFM down MEP is configured between the PE and CE. CFM detects that the CCC down and because CFM down MEP is configured, the CC messages generated have the RDI bit. The CC messages from PE2 to CE2 have the RDI bit set to indicate the blocked state. When PE2 becomes active, CCM down is cleared and the RDI bit is cleared from the subsequent CC messages.

Active/Active Multihoming Use case using RDI bit

Consider the topology where there are two provider edge devices (PE1 and PE2) and two customer edge devices (CE1 and CE2). PE1 is in active state while PE2 is in standby state. If CFM down MEP is not configured between the PE and CE to monitor the link connectivity, the CC messages generated do not have the RDI bit. CFM down MEP is configured between the PE and CE. CFM detects that the CCC down and because CFM down MEP is configured, the CC messages generated have the RDI bit. The CC messages from PE2 to CE2 have the RDI bit set to indicate the blocked state. When PE2 becomes active, CCM down is cleared and the RDI bit is cleared from the subsequent CC messages.

SEE ALSO

interface-status-tlv

interface-status-send-rdi

Configuring Port Status TLV and Interface Status TLV

IN THIS SECTION

- TLVs Overview | 85
- Various TLVs for CFM PDUs | 86
- Support for Additional Optional TLVs | 88
- MAC Status Defects | 96
- Configuring Remote MEP Action Profile Support | 98
- Monitoring a Remote MEP Action Profile | 99

TLVs Overview

Type, Length, and Value (TLVs) are described in the IEEE 802.1ag standard for CFM as a method of encoding variable-length and/or optional information in a PDU. TLVs are not aligned to any particular word or octet boundary. TLVs follow each other with no padding between them.

Table 1 shows the TLV format and indicates if it is required or optional.

Table 7: Format of TLVs

Parameter	Octet (sequence)	Description
Type	1	Required. If 0, no Length or Value fields follow. If not 0, at least the Length field follows the Type field.
Length	2–3	Required if the Type field is not 0. Not present if the Type field is 0. The 16 bits of the Length field indicate the size, in octets, of the Value field. 0 in the Length field indicates that there is no Value field.
Value	4	Length specified by the Length field. Optional. Not present if the Type field is 0 or if the Length field is 0.

Various TLVs for CFM PDUs

Table 8 on page 86 shows a set of TLVs defined by IEEE 802.1ag for various CFM PDU types. Each TLV can be identified by the unique value assigned to its type field. Some type field values are reserved.

Table 8: Type Field Values for Various TLVs for CFM PDUs

TLV or Organization	Type Field
End TLV	0
Sender ID TLV	1
Port Status TLV	2
Data TLV	3
Interface Status TLV	4
Reply Ingress TLV	5
Reply Egress TLV	6
LTM Egress Identifier TLV	7
LTR Egress Identifier TLV	8
Reserved for IEEE 802.1	9 to 30
Organization-Specific TLV	31
Defined by ITU-T Y.1731	32 to 63
Reserved for IEEE 802.1	64 to 255

Not every TLV is applicable for all types of CFM PDUs.

- TLVs applicable for continuity check message (CCM):
  - End TLV
  - Sender ID TLV
  - Port Status TLV
  - Interface Status TLV
  - Organization-Specific TLV
- TLVs applicable for loopback message (LBM):
  - End TLV
  - Sender ID TLV
  - Data TLV
  - Organization-Specific TLV
- TLVs applicable for loopback reply (LBR):
  - End TLV
  - Sender ID TLV
  - Data TLV
  - Organization-Specific TLV
- TLVs applicable for linktrace message (LTM):
  - End TLV
  - LTM Egress Identifier TLV
  - Sender ID TLV
  - Organization-Specific TLV
- TLVs applicable for linktrace reply (LTR):
  - End TLV
  - LTR Egress Identifier TLV
  - Reply Ingress TLV
  - Reply Egress TLV

- Sender ID TLV
- Organization-Specific TLV

The following TLVs are currently supported in the applicable CFM PDUs:

- End TLV
- Reply Ingress TLV
- Reply Egress TLV
- LTR Egress Identifier TLV
- LTM Egress Identifier TLV
- Data TLV

Support for Additional Optional TLVs

IN THIS SECTION

- Port Status TLV | **89**
- Interface Status TLV | **92**

The following additional optional TLVs are supported:

- Port Status TLV
- Interface Status TLV

MX Series routers support configuration of port status TLV and interface status TLV. Configuring the Port Status TLV allows the operator to control the transmission of the Port Status TLV in CFM PDUs.

NOTE: Although Port Status TLV configuration statements are visible in the CLI on M120 and M320 routers, Port Status TLV cannot be configured on these systems. Port Status TLV can be enabled on a MEP interface only if it is a bridge logical interface, which is not possible on these systems.

For configuration information, see the following sections:

Port Status TLV

The Port Status TLV indicates the ability of the bridge port on which the transmitting MEP resides to pass ordinary data, regardless of the status of the MAC. The value of this TLV is driven by the MEP variable `enableRmepDefect`, as shown in Table 10 on page 89. The format of this TLV is shown in Table 9 on page 89.

Any change in the Port Status TLVs value triggers one extra transmission of that bridge ports MEP CCMs.

Table 9: Port Status TLV Format

Parameter	Octet (Sequence)
Type = 2	1
Length	2–3
Value (See Table 10 on page 89)	4

Table 10: Port Status TLV Values

Mnemonic	Ordinary Data Passing Freely Through the Port	Value
psBlocked	No: `enableRmepDefect` = false	1
psUp	Yes: `enableRmepDefect` = true	2

The MEP variable `enableRmepDefect` is a boolean variable indicating whether frames on the service instance monitored by the maintenance associations if this MEP are enabled to pass through this bridge port by the Spanning Tree Protocol and VLAN topology management. It is set to TRUE if:

- The bridge port is set in a state where the traffic can pass through it.
- The bridge port is running multiple instances of the spanning tree.
- The MEP interface is not associated with a bridging domain.

Configuring Port Status TLV

Junos OS provides configuration support for the Port Status TLV, allowing you to control the transmission of this TLV in CCM PDUs. The Junos OS provides this configuration at the continuity-check level. By default, the CCM does not include the Port Status TLV. To configure the Port Status TLV, use the `port-status-tlv` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`identifier`* `maintenance-association` *`identifier`* `continuity-check]` hierarchy level.

NOTE: Port Status TLV configuration is not mandated by IEEE 802.1ag. The Junos OS provides it in order to give more flexibility to the operator; however it receives and processes CCMs with a Port Status TLV, regardless of this configuration.

An example of the configuration statements follows:

```
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain identifier {
                level number;
                    maintenance-association identifier {
                        continuity-check {
                            interval number,
                            loss-threshold number;
                            hold-interval number;
                            port-status-tlv; # Sets Port Status TLV
                        }
                    }
                }
            }
        }
    }
}
```

You cannot enable Port Status TLV transmission in the following two cases:

- If the MEP interface under the maintenance-association is not of type bridge.
- If the MEP is configured on a physical interface.

Displaying the Received Port Status TLV

The Junos OS saves the last received Port Status TLV from a remote MEP. If the received Port Status value does not correspond to one of the standard values listed in Table 10 on page 89, then the show command displays it as "unknown." You can display the last saved received Port Status TLV using the show oam ethernet connectivity-fault-management mep-database maintenance-domain *identifier* maintenance-association *identifier* local-mep *identifier* remote-mep *identifier* command, as in the following example:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md5
maintenance-association ma5 local-mep 2001 remote-mep 1001
Maintenance domain name: md5, Format: string, Level: 5
  Maintenance association name: ma5, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 2001, Direction: down, MAC address: 00:19:e2:b2:81:4a
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: up, Port status TLV: up
  Interface name: ge-2/0/0.0, Interface status: Active, Link status: Up

  Remote MEP identifier: 1001, State: ok
    MAC address: 00:19:e2:b0:74:00, Type: Learned
    Interface: ge-2/0/0.0
    Last flapped: Never
    Remote defect indication: false
    Port status TLV: none # RX PORT STATUS
    Interface status TLV: none
```

Displaying the Transmitted Port Status TLV

The Junos OS saves the last transmitted Port Status TLV from a local MEP. If the transmission of the Port Status TLV has not been enabled, then the show command displays "none." You can display the last saved transmitted Port Status TLV using the show oam ethernet connectivity-fault-management mep-database maintenance-domain *identifier* maintenance-association *identifier* local-mep *identifier* remote-mep *identifier* command, as in the following example:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md5
maintenance-association ma5 local-mep 2001 remote-mep 1001
Maintenance domain name: md5, Format: string, Level: 5
  Maintenance association name: ma5, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 2001, Direction: down, MAC address: 00:19:e2:b2:81:4a
```

```
Auto-discovery: enabled, Priority: 0
Interface status TLV: up, Port status TLV: up # TX PORT STATUS
Interface name: ge-2/0/0.0, Interface status: Active, Link status: Up

Remote MEP identifier: 1001, State: ok
  MAC address: 00:19:e2:b0:74:00, Type: Learned
  Interface: ge-2/0/0.0
  Last flapped: Never
  Remote defect indication: false
  Port status TLV: none
  Interface status TLV: none
```

Interface Status TLV

The Interface Status TLV indicates the status of the interface on which the MEP transmitting the CCM is configured, or the next-lower interface in the IETF RFC 2863 IF-MIB. The format of this TLV is shown in Table 11 on page 92. The enumerated values are shown in Table 12 on page 92.

Table 11: Interface Status TLV Format

Parameter	Octet (Sequence)
Type = 4	1
Length	2–3
Value (See Table 12 on page 92)	4

Table 12: Interface Status TLV Values

Mnemonic	Interface Status	Value
isUp	up	1
isDown	down	2
isTesting	testing	3

Table 12: Interface Status TLV Values *(Continued)*

Mnemonic	Interface Status	Value
isUnknown	unknown	4
isDormant	dormant	5
isNotPresent	notPresent	6
isLowerLayerDown	lowerLayerDown	7

NOTE: When the operational status of a logical interface changes from the down state (status value of 2) to the lower layer down state (status value of 7) and vice versa, the LinkDown SNMP trap is not generated. For example, if you configure an aggregated Ethernet interface bundle with a VLAN tag and add a physical interface that is in the operationally down state to the bundle, the operational status of the aggregated Ethernet logical interface bundle at that point is lower layer down (7). If you take the MIC associated with the interface offline, the LinkDown trap is not generated when the logical interface shifts from the lower layer down state to the down state.

Similarly, consider another sample scenario in which an physical interface is added to an aggregated Ethernet bundle that has VLAN tagging and the aggregated Ethernet logical interface is disabled. When the logical interface is disabled, the operational status of the logical interface changes to down. If you disable the physical interface that is part of the aggregated Ethernet bundle, the operational status of the aggregated Ethernet logical interface remains down. If you reenable the aggregated Ethernet logical interface, the operational status of it changes from down to lower layer down. The LinkDown SNMP trap is not generated at this point.

Configuring Interface Status TLV

The Junos OS provides configuration support for the Interface Status TLV, thereby allowing operators to control the transmission of this TLV in CCM PDUs through configuration at the continuity-check level.

NOTE: This configuration is not mandated by IEEE 802.1ag; rather it is provided to give more flexibility to the operator. The Junos OS receives and processes CCMs with the Interface Status TLV, regardless of this configuration.

The interface status TLV configuration is shown below:

```
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain identifier {
                    level number;
                    maintenance-association identifier {
                        continuity-check {
                            interval number;
                            loss-threshold number;
                            hold-interval number;
                            interface-status-tlv; # Sets the interface status TLV
                        }
                    }
                }
            }
        }
    }
}
```

NOTE: The Junos OS supports transmission of only three out of seven possible values for the Interface Status TLV. The supported values are 1, 2, and 7. However, the Junos OS is capable of receiving any value for the Interface Status TLV.

Displaying the Received Interface Status TLV

The Junos OS saves the last received Interface Status TLV from the remote MEP. If the received Interface Status value does not correspond to one of the standard values listed in Table 11 on page 92, then the `show` command displays "unknown."

You can display this last saved Interface Status TLV using the `show oam ethernet connectivity-fault-management mep-database maintenance-domain` *`identifier`* `maintenance-association` *`identifier`* `local-mep` *`identifier`* `remote-mep` *`identifier`* command, as in the following example:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md5 maintenance-association ma5 local-mep 2001 remote-mep 1001
```

```
Maintenance domain name: md5, Format: string, Level: 5
  Maintenance association name: ma5, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 2001, Direction: down, MAC address: 00:19:e2:b2:81:4a
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: up, Port status TLV: up
  Interface name: ge-2/0/0.0, Interface status: Active, Link status: Up

  Remote MEP identifier: 1001, State: ok
    MAC address: 00:19:e2:b0:74:00, Type: Learned
    Interface: ge-2/0/0.0
    Last flapped: Never
    Remote defect indication: false
    Port status TLV: none
    Interface status TLV: none # displays the Interface Status TLV state
```

Displaying the Transmitted Interface Status TLV

The Junos OS saves the last transmitted Interface Status TLV from a local MEP. If the transmission of Interface Status TLV has not been enabled, then the `show` command displays "none."

You can display the last transmitted Interface Status TLV using the `show oam ethernet connectivity-fault-management mep-database maintenance-domain` *`identifier`* `maintenance-association` *`identifier`* `local-mep` *`identifier`* `remote-mep` *`identifier`* command, as in the following example:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md5
maintenance-association ma5 local-mep 2001 remote-mep 1001

Maintenance domain name: md5, Format: string, Level: 5
  Maintenance association name: ma5, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 2001, Direction: down, MAC address: 00:19:e2:b2:81:4a
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: up, Port status TLV: up
  Interface name: ge-2/0/0.0, Interface status: Active, Link status: Up

  Remote MEP identifier: 1001, State: ok
    MAC address: 00:19:e2:b0:74:00, Type: Learned
    Interface: ge-2/0/0.0
    Last flapped: Never
    Remote defect indication: false
    Port status TLV: none
```

```
    Interface status TLV: none
```

MAC Status Defects

The Junos OS provides MAC status defect information, indicating that one or more of the remote MEPs is reporting a failure in its Port Status TLV or Interface Status TLV. It indicates “yes” if either some remote MEP is reporting that its interface is not isUp (for example, at least one remote MEPs interface is unavailable), or if all remote MEPs are reporting a Port Status TLV that contains some value other than psUp (for example, all remote MEPs Bridge Ports are not forwarding data). There are two `show` commands you can use to view the MAC Status Defects indication.

Use the `mep-database` command to display MAC status defects:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md6
maintenance-association ma6
Maintenance domain name: md6, Format: string, Level: 6
  Maintenance association name: ma6, Format: string
  Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
  MEP identifier: 500, Direction: down, MAC address: 00:05:85:73:7b:39
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: up, Port status TLV: up
  Interface name: xe-5/0/0.0, Interface status: Active, Link status: Up
  Defects:
    Remote MEP not receiving CCM           : no
    Erroneous CCM received                 : no
    Cross-connect CCM received             : no
    RDI sent by some MEP                   : no
    Some remote MEP's MAC in error state   : yes # MAC Status Defects yes/no
  Statistics:
    CCMs sent                              : 1658
    CCMs received out of sequence          : 0
    LBMs sent                              : 0
    Valid in-order LBRs received           : 0
    Valid out-of-order LBRs received       : 0
    LBRs received with corrupted data      : 0
    LBRs sent                              : 0
    LTMs sent                              : 0
    LTMs received                          : 0
    LTRs sent                              : 0
    LTRs received                          : 0
    Sequence number of next LTM request    : 0
```

```
    1DMs sent                               : 0
    Valid 1DMs received                     : 0
    Invalid 1DMs received                   : 0
    DMMs sent                               : 0
    DMRs sent                               : 0
    Valid DMRs received                     : 0
    Invalid DMRs received                   : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
      200     00:05:85:73:39:4a       ok    xe-5/0/0.0
```

Use the `interfaces` command to display MAC status defects:

```
user@host> show oam ethernet connectivity-fault-management interfaces detail
Interface name: xe-5/0/0.0, Interface status: Active, Link status: Up
  Maintenance domain name: md6, Format: string, Level: 6
    Maintenance association name: ma6, Format: string
    Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
    Interface status TLV: up, Port status TLV: up
    MEP identifier: 500, Direction: down, MAC address: 00:05:85:73:7b:39
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                  : no
      Erroneous CCM received                        : no
      Cross-connect CCM received                    : no
      RDI sent by some MEP                          : no
      Some remote MEP's MAC in error state          : yes # MAC Status Defects yes/no
    Statistics:
      CCMs sent                                     : 1328
      CCMs received out of sequence                 : 0
      LBMs sent                                     : 0
      Valid in-order LBRs received                  : 0
      Valid out-of-order LBRs received              : 0
      LBRs received with corrupted data             : 0
      LBRs sent                                     : 0
      LTMs sent                                     : 0
      LTMs received                                 : 0
      LTRs sent                                     : 0
      LTRs received                                 : 0
      Sequence number of next LTM request           : 0
      1DMs sent                                     : 0
      Valid 1DMs received                           : 0
```

```
      Invalid 1DMs received                        : 0
      DMMs sent                                    : 0
      DMRs sent                                    : 0
      Valid DMRs received                          : 0
      Invalid DMRs received                        : 0
    Remote MEP count: 1
      Identifier    MAC address        State    Interface
        200     00:05:85:73:39:4a       ok    xe-5/0/0.0
```

Configuring Remote MEP Action Profile Support

Based on values of `interface-status-tlv` and `port-status-tlv` in the received CCM packets, a specific action, such as `interface-down`, can be taken using the `action-profile` options. Multiple action profiles can be configured on the router, but only one action profile can be assigned to a remote MEP.

The action profile can be configured with at least one event to trigger the action; but the action will be triggered if any one of these events occurs. It is not necessary for all of the configured events to occur to trigger `action`.

An action-profile can be applied only at the remote MEP level.

The following example shows an action profile configuration with explanatory comments added:

```
[edit protocols oam ethernet connectivity-fault-management]
action-profile tlv-action {
    event {
        # If interface status tlv with value specified in the config is received
        interface-status-tlv down|lower-layer-down;

        # If port status tlv with value specified in the config is received
        port-status-tlv blocked;

        # If connectivity is lost to the peer */
        adjacency-loss;
    }
    action {
        # Bring the interface down */
        interface-down;
    }
    default-actions interface-down;
    }
```

```
# domains
    maintenance-domain identifier {
        # maintenance domain level (0-7)
        level number;

        # association
        maintenance-association identifier {
            mep identifier {
                interface ge-x/y/z.w;

                remote-mep identifier {
                    # Apply the action-profile for the remote MEP
                    action-profile tlv-action;
                }
            }
        }
    }
```

Monitoring a Remote MEP Action Profile

You can use the `show oam ethernet connectivity-fault-management mep-database` command to view the action profile status of a remote MEP, as in the following example:

show oam ethernet connectivity-fault- management mep-database remote-mep (Action Profile Event)

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain md5
maintenance-association ma5 remote-mep 200
Maintenance domain name: md5, Format: string, Level: 5
  Maintenance association name: ma5, Format: string
  Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
  MEP identifier: 100, Direction: down, MAC address: 00:05:85:73:e8:ad
  Auto-discovery: enabled, Priority: 0
  Interface status TLV: none, Port status TLV: none  # last status TLVs transmitted by the router
  Interface name: ge-1/0/8.0, Interface status: Active, Link status: Up

  Remote MEP identifier: 200, State: ok # displays the remote MEP name and state
    MAC address: 00:05:85:73:96:1f, Type: Configured
    Interface: ge-1/0/8.0
    Last flapped: Never
    Remote defect indication: false
    Port status TLV: none
    Interface status TLV: lower-layer-down
```

```
Action profile: juniper  # displays remote MEP's action profile identifier
  Last event: Interface-status-tlv lower-layer-down  # last remote MEP event
                                                     # to trigger action
  Action: Interface-down, Time: 2009-03-27 14:25:10 PDT (00:00:02 ago)
                          # action occurrence time
```

RELATED DOCUMENTATION

connectivity-fault-management

IEEE 802.1ag OAM Connectivity Fault Management | **21**

Configuring Chassis ID TLV

In Release 16.1R2 and later, you can configure Junos OS to send the sender ID TLV along with the packets. The sender ID TLV is an optional TLV that is sent in continuity check messages (CCMs), loopback messages, and Link Trace Messages (LTMs), as specified in the IEEE 802.1ag standard. The sender ID TLV contains the chassis ID, which is the unique, CFM-based MAC address of the device, and the management IP address, which is an IPv4 or an IPv6 address.

The value of the `length` field in the TLV indicates whether or not the TLV contains the chassis ID information. The possible values for the `length` field are zero (`0`) or any valid number, which indicates the absence or presence of chassis ID information in the TLV, respectively.

You can enable Junos OS to send the sender ID TLV at the global level by using the `set protocols oam ethernet connectivity-fault-management sendid-tlv send-chassis-tlv` command. If the sender ID TLV is configured at the global level, then the default maintenance domain, maintenance association, and the maintenance association intermediate point (MIP) half function inherit this configuration.

You can also configure the sender ID TLV at the following hierarchy levels:

- `[edit protocols oam ethernet connectivity-fault-management]`.
- `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`maintenance-domain-name`* `maintenance-association` *`maintenance-association-name`* `continuity-check]`.

The sender ID TLV configuration at the maintenance-association level takes precedence over the global-level configuration.

NOTE: The sender ID TLV is supported only for 802.1ag PDUs and is not supported for performance monitoring protocol data units (PDUs).

SEE ALSO

IEEE 802.1ag OAM Connectivity Fault Management | 21

Configuring MAC Flush Message Processing in CET Mode

IN THIS SECTION

- Configuring a Connection Protection TLV Action Profile | 103

In carrier Ethernet transport (CET) mode, MX Series routers are used as provider edge (PE) routers, and Nokia Siemens Networks A2200 Carrier Ethernet Switches (referred to as E-domain devices) that run standard-based protocols are used in the access side. On the MX Series routers, VPLS pseudowires are configured dynamically through label distribution protocol (LDP). On the E-domain devices, topology changes are detected through connectivity fault management (CFM) sessions running between the E-domain devices and the MX Series PE routers. The MX Series PE routers can bring the carrier Ethernet interface down if there is CFM connectivity loss. This triggers a local MAC flush as well as a targeted label distribution protocol (T-LDP) MAC flush notification that gets sent towards the remote MX Series PEs to trigger MAC flush on them.

In CET inter-op mode, MX Series routers need to interoperate with the Nokia Siemens Networks Ax100 Carrier Ethernet access devices (referred to as A-domain devices) that run legacy protocols. Nokia Siemens Networks A4100 and A8100 devices act as an intermediate between the MX Series PE routers and A-domain devices. These intermediate devices perform interworking function (IWF) procedures so that operations administration management (OAM) sessions can be run between MX Series routers and A-domain devices. There are no VPLS pseudowires between the MX Series PE routers and the Nokia Siemens Networks A4100 and A8100 intermediate devices, so there is no LDP protocol running between the PE routers to send topology change notifications. In order to communicate topology changes, MX Series routers can trigger a MAC flush and propagate it in the core. MX Series routers can use action profiles based upon the connection protection type length value (TLV) event. The action profile brings down the carrier edge *logical interface* in MX Series PE routers, which will trigger a local MAC flush and also propagate the topology change to the core using LDP notification.

For VPLS there is no end-to-end connectivity monitored. The access rings are independently monitored by running CFM down multiple end points (MEPs) on the working and protection paths for each of the services between the E-domain devices and the MX Series PE routers, and between the A-domain devices and the MX Series PE routers the IWF hosted by the Nokia Siemens Networks A-4100 devices. When there is a connectivity failure on the working path, the Nokia Siemens Networks Ax200 devices perform a switchover to the protection path, triggering a topology change notification (in the form of TLVs carried in CCM) to be sent on the active path.

Figure 7: CET inter-op Dual Homed Topology

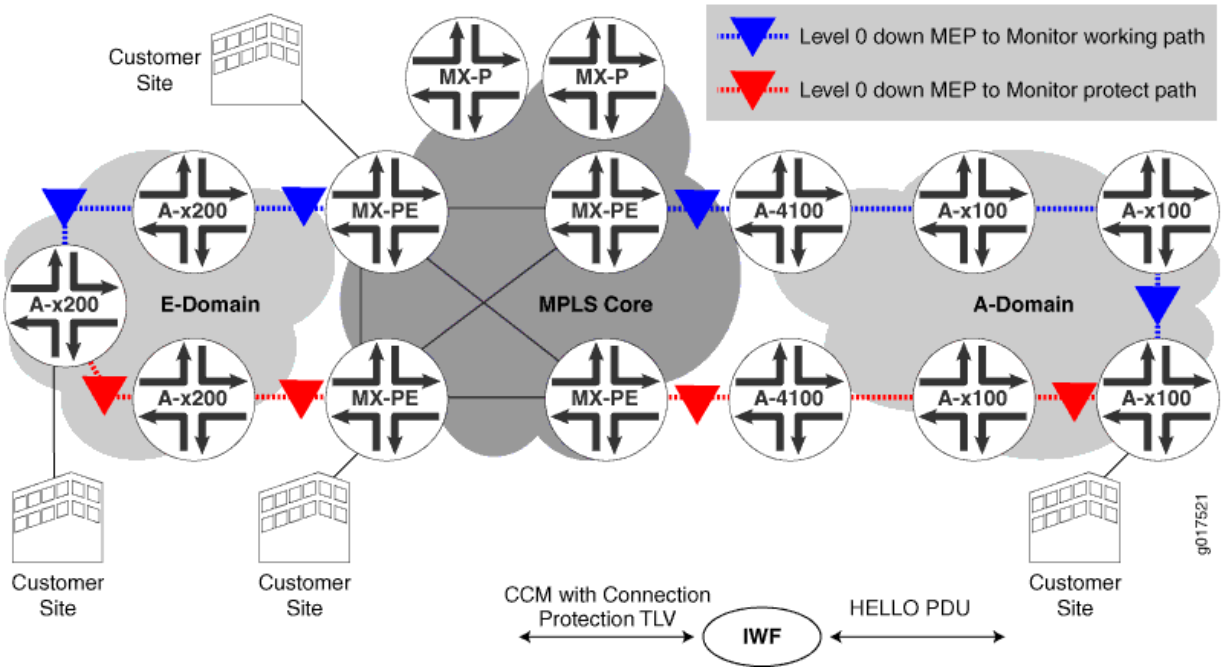


Figure 7 on page 102 describes the dual homed topology on MX Series PE routers connected to the A-domain. When an A-domain device triggers a switchover, it starts switching the service traffic to the new active path. This change is communicated in the HELLO protocol data units (PDUs) sent by that A-domain device on the working and protection paths. When the IWF in A4100 recieves these HELLO PDUs, it converts them to standard CCM messages and also inserts a connection protection TLV. The “Protection-in-use” field of the connection protection TLV is encoded with the currently active path, and is included in the CCM message. CCM messages are received by the MX Series PE routers through the VLAN spoke in A4100. In the above dual homed scenario, one MX Series PE router monitors the working path, and the other MX Series PE router monitors the protection path.

A MAC flush occurs when the CFM session that is monitoring the working path detects that the service traffic has moved to the protection path or when the CFM session that is monitoring the protection path detects that the service traffic has moved to the working path.

Figure 8: CET inter-op Dual Attached Topology

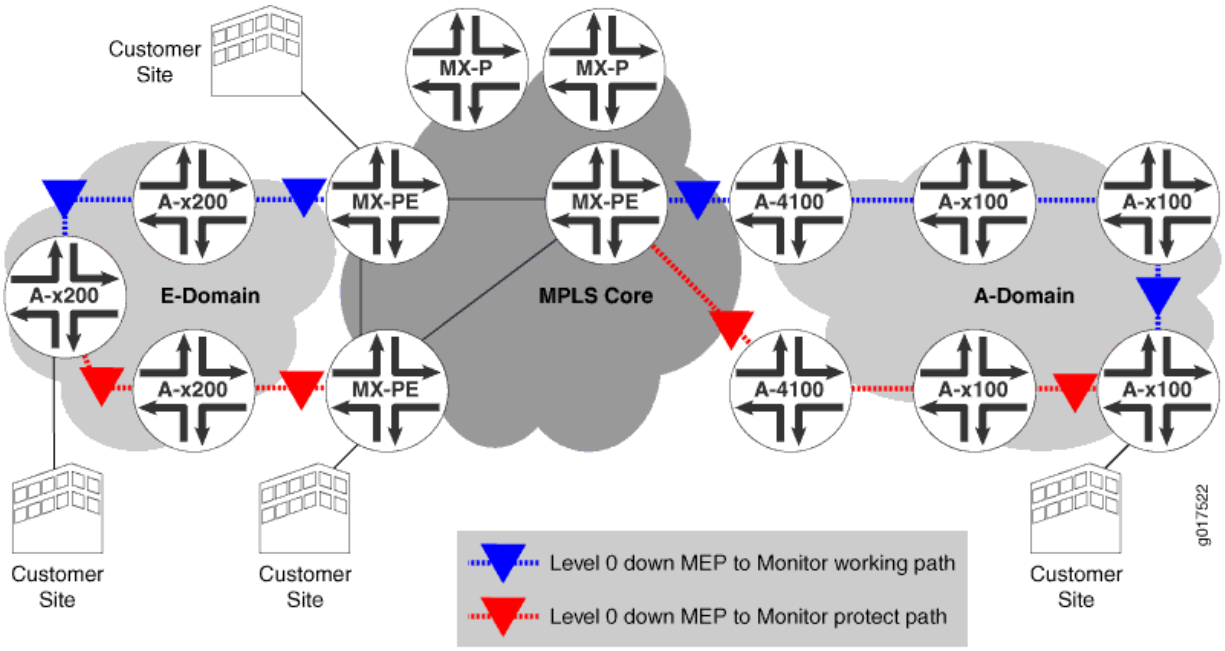


Figure 8 on page 103 describes the dual attached topology on MX Series PE routers connected to the A-domain. The MAC flush mechanism used in this case is also the same as the one used for the A-domain in the dual homed scenario (Figure 1). However in this case both the CFM sessions are hosted by only one MX Series PE router. When Ax100 in the A-domain detects topology changes, the MX Series PE router receives the connection protection TLV in the CCM message for the working and protection paths with the value of “Protection-in-use” indicating which path is the active one. Based upon the event that is generated for the CFM session, the MX Series PE router will bring down the appropriate interface which will trigger a local MAC flush.

Configuring a Connection Protection TLV Action Profile

An action profile can be configured to perform the `interface-down` action based on the values of `connection-protection-tlv` in the received CCM packets.

The following example shows an action profile configuration with explanatory comments added:

```
[edit protocols oam ethernet connectivity-fault-management]
action-profile <tlv-action> {
    event {
        # If a connection protection TLV with a “Protection-in-use” value of SET is received */
        connection-protection-tlv <using-protection-path>;
        # If a connection protection TLV with a “Protection-in-use” value of RESET is received */
        connection-protection-tlv <using-working-path>;
```

```
    }
    action {
        # Bring the interface down */
        interface-down;
    }
}
```

SEE ALSO

connection-protection-tlv

IEEE 802.1ag OAM Connectivity Fault Management | **21**

Example: Configuring an Action Profile Based on Connection Protection TLVs

IN THIS SECTION

- Requirements | **104**
- Overview and Topology | **104**
- Configuration | **105**

This example shows how to configure an action profile based on the connection protection TLV for the purposes of triggering MAC flushes based on topology changes in a CET network.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 11.2 or later
- A MX series PE router

Overview and Topology

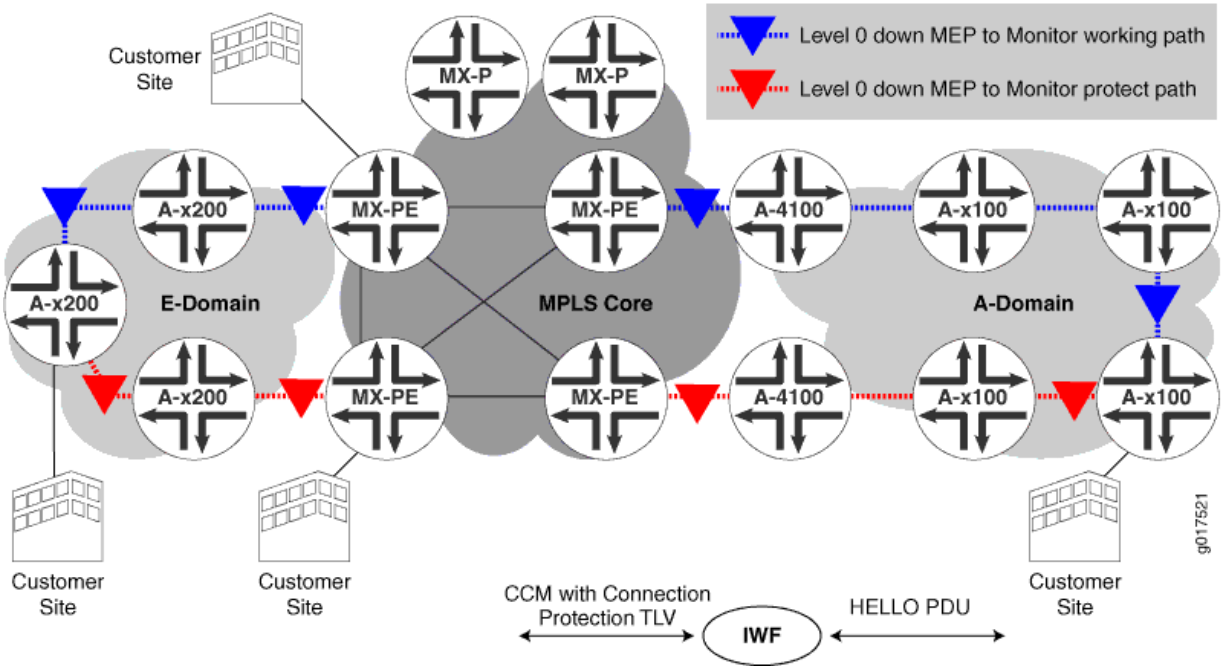
IN THIS SECTION

- Topology | **105**

The physical topology of a CET network using MX series PE routers is shown in Figure 9 on page 105

Topology

Figure 9: Topology of CET network



The following definitions describe the meaning of the device abbreviation and terms used in Figure 9 on page 105.

- Provider edge (PE) device—A device, or set of devices, at the edge of the provider network that presents the provider's view of the customer site.
- E-domain—Nokia Siemens Networks Carrier Ethernet Switches that run standard based protocols and are used in the access side.
- A-domain—Nokia Siemens Networks Carrier Ethernet Switches that run legacy protocols.

Configuration

IN THIS SECTION

- Procedure | **106**

Procedure

Step-by-Step Procedure

To configure an action profile based on the connection protection TLV, preform these tasks:

1. Configure an action profile

```
[edit protocols oam ethernet connectivity-fault-management]
action-profile <tlv-action> {
    event {
```

2. If the connection protection TLV is received with a “Protection-in-use” value of SET, then the connection protection TLV should use the protection path

```
connection-protection-tlv <using-protection-path>;
```

3. If the connection protection TLV is received with a “Protection-in-use” value of RESET, then the connection protection TLV should use the working path

```
connection-protection-tlv <using-working-path>;
}
```

4. Configure the action profile to bring the interface down

```
action {
    /* Bring the interface down */
    interface-down;
}
}
```

Results

Check the results of the configuration

```
[edit protocols oam ethernet connectivity-fault-management]
action-profile <tlv-action> {
    event {
```

```
        connection-protection-tlv <using-protection-path>;
        connection-protection-tlv <using-working-path>;
    }
    action {
        interface-down;
    }
}
```

SEE ALSO

connection-protection-tlv

Release History Table

Release	Description
17.3R1	Starting in Junos OS Release 17.3R1, you can enable connectivity fault management (CFM) monitoring between provider edge devices and customer edge devices when the customer edge device is not a Juniper device by using the remote defect indication (RDI) bit.
16.1	In Release 16.1R2 and later, you can configure Junos OS to send the sender ID TLV along with the packets.

RELATED DOCUMENTATION

Introduction to OAM Connectivity Fault Management (CFM) | 19

ITU-T Y.1731 Ethernet Service OAM Overview | 204

Configure Continuity Check Messages

IN THIS SECTION

- Configure Faster Protection Switching for Point-to-Point Network Topologies | 108
- Configure Faster Convergence for Dual-Homed Multipoint-to-Multipoint Network Topologies | 110
- Configure a Primary VLAN ID for Increased Flexibility | 111

- Configure a Remote Maintenance Association to Accept a Different ID | 112

Junos OS provides enhancements to trigger faster protection-switching and convergence in the event of failures in Ethernet domains for Carrier Ethernet services. These enhancements can be used when CE devices in the Ethernet domain detect faster service failures and propagates the information in the interface-status TLV of the continuity-check messages (CCMs). When CCMs are received, PE devices can perform certain actions which facilitates faster protection-switching and convergence. You can configure CCM for better scalability using the information provided in this topic.

Configure Faster Protection Switching for Point-to-Point Network Topologies

You can apply an action profile to provide faster protection switching for point-to-point network topologies with local switching configured. In a normal state, CCM sessions are configured on the working and protect interfaces. The CCM packets transmitted contain an interface-status TLV with the value up on the working interface and value down on the protect interface. When a link fails on the working interface, the protect interface starts receiving the interface-status TLV as up. With the profile configuration, if the interface-status TLV received on the protect interface is up, the working interface is automatically marked as `interface-down`.

To configure the `interface-status-tlv` down event, include the `interface-status-tlv down` statement at the `[edit protocols oam ethernet connectivity-fault-management action-profile` *`profile-name`* `event]` hierarchy level.

To configure `interface-down` as the action profile's action, include the `interface-down` statement at the `[edit protocols oam ethernet connectivity-fault-management action-profile` *`profile-name`* `action]` hierarchy level.

To configure *peer-interface* as the clear-action, include *peer-interface* at the `[edit protocols oam ethernet connectivity-fault-management action-profile` *`profile-name`* `clear-action]` hierarchy level.

```
[edit protocols oam]
ethernet  {
    connectivity-fault-management   {
        action-profile p1 {
            event {
                interface-status-tlv down;
            }
            action {
                interface-down;
            }
            clear-action {
                interface-down peer-interface;
```

```
            }
        }
    }
}
```

In this action profile configuration, when the interface-status TLV is received as up, the *peer-interface* is marked as down.

The *peer-interface* is configured in the `protect-maintenance-association` statement. Consider the following example using the `protect-maintenance-association` statement in the configuration:

```
[edit protocols oam]
ethernet  {
    connectivity-fault-management   {
        action-profile p1 {
            event {
                adjacency-loss;
            }
            action {
                interface-down;
            }
            clear-action {
                interface-down peer-interface;
            }
        }
        maintenance-domain nsn {
            level 5;
            maintenance-association ma1 {
                protect-maintenance-association ma2;
                continuity-check {
                    interval 100ms;
                    connection-protection-tlv;
                }
                mep 100 {
                    interface ge-1/1/0.0;
                    direction down;
                    auto-discovery;
                }
            }
            maintenance-association ma2 {
                continuity-check {
                    interval 100ms;
```

```
                    connection-protection-tlv;
            }
            mep 101 {
                    interface ge-1/2/0.0;
                    direction down;
                    auto-discovery;
            }
            remote-mep 100
                action-profile p1;
            }
        }
    }
}
```

SEE ALSO

> *connectivity-fault-management*

Configure Faster Convergence for Dual-Homed Multipoint-to-Multipoint Network Topologies

You can apply an action profile to provide faster convergence for dual-homed multipoint-to-multipoint network topologies. If a multipoint-to-multipoint Ethernet service uses MAC-based forwarding and stale MAC addresses exist in the learning tables, this can result in traffic black holes in the network where incoming traffic is silently discarded, without informing the source that the data did not reach its intended recipient. With the profile configuration, if the interface-status TLV received on the protect interface is up, then the interface-status TLV on the working interface is marked as down and the PE device for the protect interface propagates a remote MAC-flush message to the PE devices in the virtual private LAN service (VPLS) by using TLDP-MAC-FLUSH. The MAC flush avoids null-route filtering due to stale mac-db entries.

To configure the `interface-status-tlv` down event, include the `interface-status-tlv down` statement at the `[edit protocols oam ethernet connectivity-fault-management action-profile` *`profile-name`* `event]` hierarchy level.

To configure `propagate-remote-flush` as the action profile's action, include the `propagate-remote-flush` statement at the `[edit protocols oam ethernet connectivity-fault-management action-profile` *`profile-name`* `action]` hierarchy level.

To configure `propagate-remote-flush` as the clear-action, include the `propagate-remote-flush` statement at the `[edit protocols oam ethernet connectivity-fault-management action-profile` *`profile-name`* `clear-action]` hierarchy level.

```
[edit protocols oam]
ethernet  {
    connectivity-fault-management   {
        action-profile test {
            event {
                interface-status-tlv down;
            }
            action {
                propagate-remote-mac-flush;
            }
            clear-action {
                propagate-remote-mac-flush;
            }
        }
    }
}
```

In this action profile configuration, when the incoming CCM packet contains the interface-status TLV with value down, the `propagate-remote-mac-flush` action is triggered for the action-profile.

SEE ALSO

> IEEE 802.1ag OAM Connectivity Fault Management | **21**
>
> *connectivity-fault-management*

Configure a Primary VLAN ID for Increased Flexibility

You can assign a primary virtual LAN (VLAN) ID in the maintenance association for increased flexibility in the number of tags. When a `vlan-range` or `vlan-id-list` is configured on an interface, the service OAM must run on one of the VLANs. The VLAN assigned for service monitoring is considered the primary VLAN. If a `primary-vid` is not configured, Junos OS assigns the first VLAN from the `vlan-range` or `vlan-id-list`. In earlier releases, Junos OS assigned VLAN 4095.

To configure a primary VLAN ID, you can specify the `primary-vid` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`*`]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management]
maintenance domain md3 {
    level 3;
    maintenance-association ma3 {
        primary-vid 2000;
        continuity-check {
            interval 10ms;
            connection-protection-tlv;
        }
        mep 2 {
            interface ge-2/2/0.0;
            direction up;
            auto-discovery;
        }
    }
}
```

SEE ALSO

> IEEE 802.1ag OAM Connectivity Fault Management | **21**
>
> *connection-protection-tlv*
>
> *connectivity-fault-management*

Configure a Remote Maintenance Association to Accept a Different ID

You can configure a maintenance association to accept a different maintenance association identifier (ID) from a neighbor by including a `remote-maintenance-association` statement. The 802.1ag CCM sessions expect the same maintenance association identifier from its neighbors. If there is a maintenance association identifier mismatch, the PDUs are marked as error PDUs. If a `remote-maintenance-association` statement is configured, a different maintenance association identifier is accepted and the 802.1ag CCM sessions do not mark the CCM PDUs as error PDUs when the maintenance-association name is the same as the name specified in the `remote-maintenance-association` statement.

To configure a remote maintenance association, include the `remote-maintenance-association` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`*`]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management]
maintenance domain md3 {
    level 1;
    maintenance-association ma3 {
        remote-maintenance-association fix-ma;
        continuity-check {
            interval 10ms;
            connection-protection-tlv;
        }
        mep 2 {
            interface ge-2/2/0.0;
            direction up;
            auto-discovery;
        }
    }
}
```

Using this configuration, interoperability is improved for CCMs with low-end CE devices supporting fixed maintenance association identifier configurations.

SEE ALSO

IEEE 802.1ag OAM Connectivity Fault Management | **21**

connectivity-fault-management

connection-protection-tlv

RELATED DOCUMENTATION

Introduction to OAM Connectivity Fault Management (CFM) | **19**

Configure Connectivity Fault Management (CFM) | **26**

Example: Configure Ethernet CFM on Physical Interfaces

IN THIS SECTION

- Requirements | 114
- Overview | 114
- Configuration | 114

This example shows the configuration of Ethernet connectivity fault management (CFM) on physical interfaces.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 9.3 or later.

Overview

CFM can be used to monitor the physical link between two routers. This functionality is similar to that supported by the IEEE 802.3ah LFM protocol.

In Junos OS Release 9.3 and later, CFM also supports aggregated Ethernet interfaces. On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series routers, CFM is not supported on untagged aggregated Ethernet member links. MPCs and MICs do support CFM on untagged and tagged aggregated Ethernet logical interfaces.

NOTE: The configurations in this example are only partial examples of complete and functional router configurations. Do not copy these configurations and use them directly on an actual system.

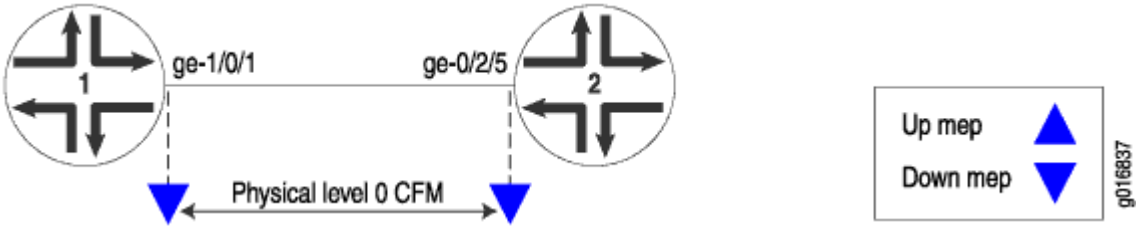
Configuration

IN THIS SECTION

- CLI Quick Configuration | 115

In the following example, two routers (Router 1 and Router 2) are connected by a point-to-point Gigabit Ethernet link. The link between these two routers is monitored using CFM. This is shown in Figure 10 on page 115. The single boundary is a “down mep” in CFM terminology.

Figure 10: Ethernet CFM on Physical Interfaces



To configure Ethernet CFM on physical interfaces, perform these tasks:

CLI Quick Configuration

Router 1

Configure the interface and CFM:

```
[edit]
interfaces ge-1/0/1 {
    unit 0 {
        family inet;
    }
}

protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain private {
                    level 0;
                    maintenance-association private-ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 100 {
                            interface ge-1/0/1;
                            direction down;
                            auto-discovery;
```

```
                        }
                    }
                }
            }
        }
    }
}
```

The configuration on Router 2 mirrors that on Router 1, with the exception of the *mep-id*.

Router 2

Configure the interface and CFM:

```
[edit]
interfaces ge-0/2/5 {
    unit 0 {
        family inet;
    }
}

protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain private {
                    level 0;
                    maintenance-association private-ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 200 {
                            interface ge-0/2/5;
                            direction down;
                            auto-discovery;
                        }
                    }
                }
            }
        }
    }
}
```

To verify that the physical interface is configured correctly for CFM, use the `show interface` command. To verify the CFM configuration, use one or more of the `show oam ethernet connectivity-fault-management` commands listed in the CLI Explorer.

RELATED DOCUMENTATION

show oam ethernet connectivity-fault-management interfaces

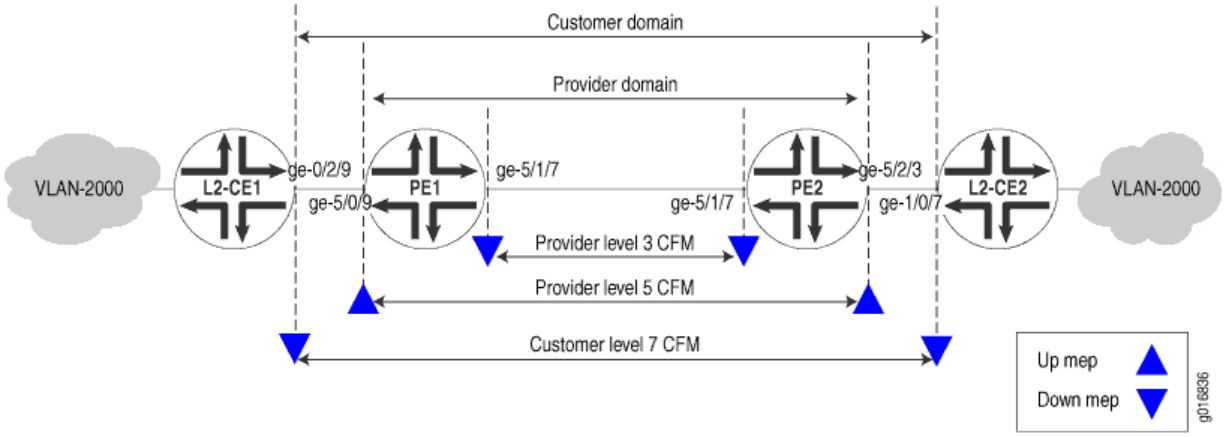
Example: Configure Ethernet CFM on Bridge Connections

In this example, both the customer and service provider are running Ethernet CFM over a simple bridge network. The network is shown in Figure 11 on page 117. The customer has configured Ethernet CFM on MX Series routers L2-CE1 and L2-CE2. The service provider has configured Ethernet CFM on MX Series routers PE1 and PE2.

NOTE: The configurations in this example are only partial examples of complete and functional router configurations. Do not copy these configurations and use them directly on an actual system.

The service provider is using CFM level 3 for the link between PE1 and PE2 and level 5 from one CE facing port to the other. The customer is using CFM level 7. The boundaries are marked with “up mep” and “down mep” CFM terminology in the figure.

Figure 11: Ethernet CFM over a Bridge Network



Here are the configurations of CFM on the customer routers.

CFM on L2-CE1

```
[edit interfaces]
ge-0/2/9 {
    vlan-tagging;
    unit 0 {
        vlan-id 2000;
    }
}

[edit protocols oam ethernet]
connectivity-fault-management {
    maintenance-domain customer {
        level 7;
        maintenance-association customer-site1 {
            continuity-check {
                interval 1s;
            }
            mep 700 {
                interface ge-0/2/9.0;
                direction down;
                auto-discovery;
            }
        }
    }
}
```

CFM on L2-CE2

```
[edit interfaces]
ge-1/0/7 {
    vlan-tagging;
    unit 0 {
        vlan-id 2000;
    }
}

[edit protocols oam ethernet]
connectivity-fault-management {
    maintenance-domain customer {
        level 7;
```

```
        maintenance-association customer-site2 {
            continuity-check {
                interval 1s;
            }
            mep 800 {
                interface ge-1/0/7.0;
                direction down;
                auto-discovery;
            }
        }
    }
}
```

Here are the configurations of CFM on the provider routers.

CFM on PE1

```
[edit interfaces]
ge-5/0/9 {
    vlan-tagging;
    encapsulation flexible-ethernet-services;
    unit 0 {
        encapsulation vlan-bridge;
        vlan-id 2000;
    }
}
ge-5/1/7 {
    vlan-tagging;
    encapsulation flexible-ethernet-services;
    unit 0 {
        encapsulation vlan-bridge;
        vlan-id 2000;
    }
}

[edit bridge-domains]
bridge-vlan2000 {
    domain-type bridge;
    vlan-id 2000;
    interface ge-5/0/9.0;
    interface ge-5/1/7.0;
}
```

```
[edit protocols oam ethernet connectivity-fault-management]
maintenance-domain provider-outer {
    level 5;
    maintenance-association provider-outer-site1 {
        continuity-check {
            interval 1s;
        }
        mep 200 {
            interface ge-5/0/9.0;
            direction up;
            auto-discovery;
        }
    }
}
maintenance-domain provider-inner {
    level 3;
    maintenance-association provider-inner-site1 {
        continuity-check {
            interval 1s;
        }
        mep 200 {
            interface ge-5/1/7.0;
            direction down;
            auto-discovery;
        }
    }
}
```

CFM on PE2

```
[edit interfaces]
ge-5/1/7 {
    vlan-tagging;
    encapsulation flexible-ethernet-services;
    unit 0 {
        encapsulation vlan-bridge;
        vlan-id 2000;
    }
}
ge-5/2/3 {
    vlan-tagging;
```

```
    encapsulation flexible-ethernet-services;
    unit 0 {
        encapsulation vlan-bridge;
        vlan-id 2000;
    }
}

[edit bridge-domains]
bridge-vlan2000 {
    domain-type bridge;
    interface ge-5/2/3.0;
    interface ge-5/1/7.0;
}

[edit protocols oam ethernet connectivity-fault-management]
maintenance-domain provider-outer {
    level 5;
    maintenance-association provider-outer-site1 {
        continuity-check {
            interval 1s;
        }
        mep 100 {
            interface ge-5/2/3.0;
            direction up;
            auto-discovery;
        }
    }
}
maintenance-domain provider-inner {
    level 3;
    maintenance-association provider-inner-site1 {
        continuity-check {
            interval 1s;
        }
        mep 100 {
            interface ge-5/1/7.0;
            direction down;
            auto-discovery;
        }
    }
}
```

RELATED DOCUMENTATION

Configure Continuity Check Messages | 107

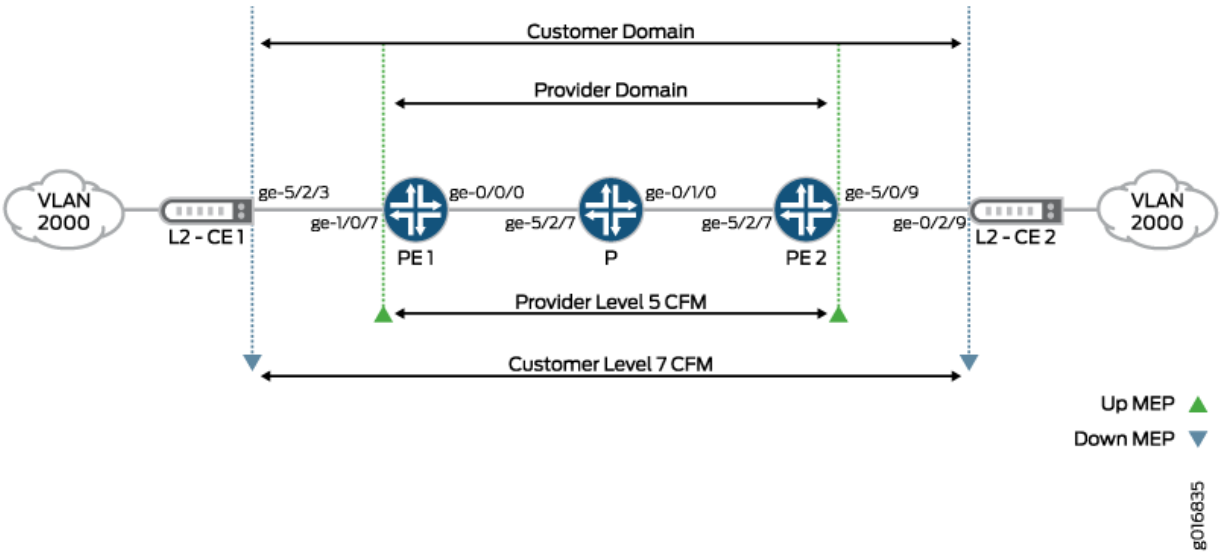
Example: Configure Ethernet CFM over VPLS

In this example, both the customer and service provider are running Ethernet CFM over a VPLS and a multiprotocol label switching (MPLS) network. The network is shown in Figure 12 on page 122. The customer has configured Ethernet CFM on MX Series routers L2-CE1 and L2-CE2. The service provider has configured Ethernet CFM on MX Series routers PE1, P, and PE2.

NOTE: The configurations in this example are only partial examples of complete and functional router configurations. Do not copy these configurations and use them directly on an actual system.

The service provider is using CFM level 5 and the customer is using CFM level 7. The boundaries are marked with “up mep” and “down mep” CFM terminology in the figure.

Figure 12: Ethernet OAM with VPLS



NOTE: The logical interfaces in a VPLS routing instance might have the same or different VLAN configurations. VLAN normalization is required to switch packets correctly among these interfaces. Normalization supports automatic mapping of VLANs and performs operations on VLAN tags to achieve the desired translation. See Configuring a Normalized VLAN for Translation or Tagging.

NOTE: The following forwarding path considerations must be observed:

- Packet receive path:
  - This is the forwarding path for packets received on the interfaces.
  - 802.1ag Ethernet OAM for VPLS uses implicit interface filters and forwarding table filters to flood, accept, and drop the CFM packets.
- Packet transmit path:
  - Junos OS uses the router's hardware-based forwarding for CPU-generated packets.
  - For down MEPs, the packets are transmitted on the interface on which the MEP is configured.
  - In MX series routers, for up MEPs, the packets must be flooded to other interfaces in the VPLS routing instance. The router creates a flood route tied to a flood next hop (with all interfaces to flood) and then sources the packets to be forwarded with this flood route.

The following are the configurations of the VPLS and CFM on the service provider routers.

Configuration of PE1

```
[edit chassis]
fpc 5 {
    pic 0 {
        tunnel-services {
            bandwidth 1g;
        }
    }
}

[edit interfaces]
ge-1/0/7 {
```

```
    encapsulation flexible-ethernet-services;
    vlan-tagging;
    unit 1 {
        encapsulation vlan-vpls;
        vlan-id 2000;
    }
}
ge-0/0/0 {
    unit 0 {
        family inet {
            address 10.200.1.1/24;
        }
        family mpls;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.168.231/32 {
                primary;
            }
            address 127.0.0.1/32;
        }
    }
}

[edit routing-instances]
vpls-vlan2000 {
    instance-type vpls;
    vlan-id 2000;
    interface ge-1/0/7.1;
    route-distinguisher 10.255.168.231:2000;
    vrf-target target:1000:1;
    protocols {
        vpls {
            site-range 10;
            site vlan2000-PE1 {
                site-identifier 2;
            }
        }
    }
}
```

```
[edit protocols]
rsvp {
    interface ge-0/0/0.0;
}
mpls {
    label-switched-path PE1-to-PE2 {
        to 10.100.1.1;
    }
    interface ge-0/0/0.0;
}
bgp {
    group PE1-to-PE2 {
        type internal;
        local-address 10.200.1.1;
        family l2vpn {
            signaling;
        }
        local-as 65000;
        neighbor 10.100.1.1;
    }
}
ospf {
    traffic-engineering;
    reference-bandwidth 4g;
    area 0.0.0.0 {
        interface all;
        interface fxp0.0 {
            disable;
        }
        interface ge-0/0/0.0;
    }
}
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain customer-site1 {
                level 5;
                maintenance-association customer-site1 {
                    continuity-check {
                        interval 1s;
                    }
                    mep 100 {
                        interface ge-1/0/7.1;
```

```
                        direction up;
                        auto-discovery;
                    }
                }
            }
        }
    }
}
```

Configuration of PE2

```
[edit chassis]
fpc 5 {
    pic 0 {
        tunnel-services {
            bandwidth 1g;
        }
    }
}

[edit interfaces]
ge-5/0/9 {
    vlan-tagging;
    encapsulation flexible-ethernet-services;
    unit 1 {
        encapsulation vlan-vpls;
        vlan-id 2000;
    }
}
ge-5/2/7 {
    unit 0 {
        family inet {
            address 10.100.1.1/24;
        }
        family mpls;
    }
}
lo0 {
    unit 0 {
        family inet {
            address 10.255.168.230/32 {
                primary;
```

```
            }
            address 127.0.0.1/32;
        }
    }
}

[edit routing-instances]
vpls-vlan2000 {
    instance-type vpls;
    vlan-id 2000;
    interface ge-5/0/9.1;
    route-distinguisher 10.255.168.230:2000;
    vrf-target target:1000:1;
    protocols {
        vpls {
            site-range 10;
            site vlan2000-PE2 {
                site-identifier 1;
            }
        }
    }
}

[edit protocols]
rsvp {
    interface ge-5/2/7.0;
}
mpls {
    label-switched-path PE2-to-PE1 {
        to 10.200.1.1;
    }
    interface ge-5/2/7.0;
}
bgp {
    group PE2-to-PE1 {
        type internal;
        local-address 10.100.1.1;
        family l2vpn {
            signaling;
        }
        local-as 65000;
        neighbor 10.200.1.1;
    }
```

```
}
ospf {
    traffic-engineering;
    reference-bandwidth 4g;
    area 0.0.0.0 {
        interface all;
        interface fxp0.0 {
            disable;
        }
        interface ge-5/2/7.0;
    }
}
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain customer-site1 {
                level 5;
                maintenance-association customer-site1 {
                    continuity-check {
                        interval 1s;
                    }
                    mep 200 {
                        interface ge-5/0/9.1;
                        direction up;
                        auto-discovery;
                    }
                }
            }
        }
    }
}
```

Configuration of P router

MPLS only, no CFM needed:

```
[edit]
interfaces {
    ge-5/2/7 {
        # Connected to PE1
        unit 0 {
            family inet {
```

```
                address 10.200.1.10/24;
            }
            family mpls;
        }
    }
    ge-0/1/0 {
        # Connected to PE2
        unit 0 {
            family inet {
                address 10.100.1.10/24;
            }
            family mpls;
        }
    }
    lo0 {
        unit 0{
            family inet {
                address 10.255.168.240/32;
            }
        }
    }
}

[edit]
protocols {
    rsvp {
        interface ge-0/1/0.0;
        interface ge-5/2/7.0;
    }
    mpls {
        interface ge-0/1/0.0;
        interface ge-5/2/7.0;
    }
    ospf {
        traffic-engineering;
        reference-bandwidth 4g;
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
            interface ge-0/1/0.0;
            interface ge-5/2/7.0;
```

```
        }
    }
}
```

CFM on L2-CE1

Here is the configuration of CFM on L2-E1:

```
[edit interfaces]
ge-5/2/3 {
    vlan-tagging;
    unit 0 {
        vlan-id 2000;
    }
}

[edit protocols oam]
ethernet {
    connectivity-fault-management {
        maintenance-domain customer {
            level 7;
            maintenance-association customer-site1 {
                continuity-check {
                    interval 1s;
                }
                mep 800 {
                    interface ge-5/2/3.0;
                    direction down;
                    auto-discovery;
                }
            }
        }
    }
}
```

CFM on L2-CE2

Here is the configuration of CFM L2-CE2:

```
[edit interfaces]
ge-0/2/9 {
    vlan-tagging;
```

```
    unit 0 {
        vlan-id 2000;
    }
}

[edit protocols oam]
ethernet {
    connectivity-fault-management {
        maintenance-domain customer {
            level 7;
            maintenance-association customer-site1 {
                continuity-check {
                    interval 1s;
                }
                mep 700 {
                    interface ge-0/2/9.0;
                    direction down;
                    auto-discovery;
                }
            }
        }
    }
}
```

RELATED DOCUMENTATION

Configure Continuity Check Messages | 107

CHAPTER 2

Link Fault Management for Routers

IN THIS CHAPTER

- Introduction to OAM Link Fault Management (LFM) | 132
- Configure Link Fault Management | 137
- Remote Fault Detection for Link Fault Management | 155
- Remote Loopback for Link Fault Management | 158

Introduction to OAM Link Fault Management (LFM)

SUMMARY

This section describes the Operation, Administration, and Management (OAM) of link fault management (LFM).

IN THIS SECTION

- IEEE 802.3ah OAM Link-Fault Management Overview | 132
- Understanding Ethernet OAM Link Fault Management for ACX Series Routers | 133
- Configuring Ethernet 802.3ah OAM | 135

IEEE 802.3ah OAM Link-Fault Management Overview

Ethernet interfaces capable of running at 100 Mbps or faster on EX Series switches, PTX Series, MX Series, M Series (except M5 and M10 routers), and T Series routers support the IEEE 802.3ah standard for Operation, Administration, and Management (OAM). You can configure IEEE 802.3ah OAM on Ethernet point-to-point direct links or links across Ethernet repeaters. The IEEE 802.3ah standard meets the requirement for OAM capabilities as Ethernet moves from being solely an enterprise technology to being a WAN and access technology, as well as being backward-compatible with existing Ethernet technology. Junos OS supports IEEE 802.3ah link-fault management.

The features of link-fault management are:

- Discovery
- Link monitoring
- Remote fault detection
- Remote loopback

Starting in Junos OS Release 17.3R1, the Ethernet link fault management daemon (lfmd) runs on the backup Routing Engine as well when graceful Routing Engine switchover (GRES) is configured.

The following features are not supported:

- Ethernet running on top of a Layer 2 protocol, such as Ethernet over ATM, is not supported in OAM configurations.
- Remote loopback is not supported on the 10-Gigabit Ethernet LAN/WAN PIC with SFP+.
- The remote loopback feature mentioned in section 57.2.11 of IEEE 802.3ah is not supported on T4000 routers.

NOTE: Aggregated Ethernet member links will now use the physical MAC address as the source MAC address in 802.3ah OAM packets.

Understanding Ethernet OAM Link Fault Management for ACX Series Routers

The Juniper Networks Junos operating system (Junos OS) for Juniper Networks ACX Series routers allows the Ethernet interfaces on these routers to support the IEEE 802.3ah standard for the Operation, Administration, and Maintenance (OAM) of Ethernet in access networks. The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters. The IEEE 802.3ah standard meets the requirement for OAM capabilities even as Ethernet moves from being solely an enterprise technology to a WAN and access technology, and the standard remains backward compatible with the existing Ethernet technology.

Ethernet OAM provides tools that network management software and network managers can use to determine how a network of Ethernet links is functioning. Ethernet OAM should:

- Rely only on the media access control (MAC) address or virtual LAN identifier for troubleshooting.
- Work independently of the actual Ethernet transport and function over physical Ethernet ports or a virtual service such as a pseudowire.
- Isolate faults over a flat (or single-operator) network architecture or nested or hierarchical (or multiprovider) networks.

The following OAM LFM features are supported on ACX Series routers:

- Discovery and Link Monitoring

  The discovery process is triggered automatically when OAM is enabled on the interface. The discovery process permits Ethernet interfaces to discover and monitor the peer on the link if it also supports the IEEE 802.3ah standard. You can specify the discovery mode used for IEEE 802.3ah OAM support. In active mode, the interface discovers and monitors the peer on the link if the peer also supports IEEE 802.3ah OAM functionality. In passive mode, the peer initiates the discovery process. After the discovery process has been initiated, both sides participate in the process. The router performs link monitoring by sending periodic OAM protocol data units (PDUs) to advertise OAM mode, configuration, and capabilities.

  You can specify the number of OAM PDUs that an interface can skip before the link between peers is considered down.

- Remote Fault Detection

  Remote fault detection uses flags and events. Flags are used to convey the following:

  - **Link Fault** means a loss of signal
  - **Dying Gasp** means an unrecoverable condition such as a power failure. In this condition, the local peer informs the remote peer about the failure state. When the remote peer receives a dying-gasp PDU, it takes an action corresponding to the action profile configured with the **link-adjacency-loss** event.

    NOTE: ACX5096 and ACX5048 routers do not support dying-gasp.

    When LFM is configured on an interface, a dying-gasp PDU is generated for the interface on the following failure conditions:

    - Power failure
    - Packet Forwarding Engine panic or a crash
  - **Critical Event** means an unspecified vendor-specific critical event.

  You can specify the interval at which OAM PDUs are sent for fault detection.

  NOTE: ACX Series routers support the receipt of dying-gasp packets, but cannot generate them.

- Remote Loopback Mode

Remote loopback mode ensures link quality between the router and a remote peer during installation or troubleshooting. In this mode, when the interface receives a frame that is not an OAM PDU or a PAUSE frame, it sends it back on the same interface on which it was received. The link appears to be in the active state. You can use the returned loopback acknowledgement to test delay, *jitter*, and throughput.

If a remote data terminal equipment (DTE) supports remote loopback mode, Junos OS can place the remote DTE into loopback mode. When you place a remote DTE into loopback mode, the interface receives the remote loopback request and puts the interface into remote loopback mode. When the interface is in remote loopback mode, all frames except OAM PDUs and PAUSE frames are looped back. No changes are made to the frames. OAM PDUs continue to be sent and processed.

Configuring Ethernet 802.3ah OAM

The IEEE 802.3ah standard for Operation, Administration, and Management (OAM) provides a specification for *Ethernet in the first mile (EFM)* connectivity. EFM defines how Ethernet can be transmitted over new media types using new Ethernet physical layer (PHY) interfaces. You can configure IEEE 802.3ah OAM on Ethernet point-to-point direct links or links across Ethernet repeaters. The IEEE 802.3ah OAM standard meets the requirement for OAM capabilities as Ethernet moves from being solely an enterprise technology to being a WAN and access technology, as well as being backward-compatible with existing Ethernet technology.

For Ethernet interfaces capable of running at 100 Mbps or faster, the IEEE 802.3ah OAM standard is supported on numerous Juniper Networks routers and switches. This topic describes configuration support for IEEE 802.3ah OAM features on routers.

Beginning in Junos OS Release 12.1, PTX Series routers support the following IEEE 802.3ah OAM features at the physical interface level:

- Discovery and link monitoring
- Fault signaling and detection
- Periodic packet management (PPM) processing
- Action profile support
- Graceful Routing Engine switchover (GRES)

To configure 802.3ah OAM support for Ethernet interfaces, include the `oam` statement at the `[edit protocols]` hierarchy level:

```
oam {
    ethernet {
        link-fault-management {
            interfaces {
```

```
                interface-name {
                    pdu-interval interval;
                    link-discovery (active | passive);
                    pdu-threshold count;
                }
            }
        }
    }
}
```

You can configure threshold values for fault events that trigger the sending of link event TLVs when the values exceed the threshold. To set threshold values for fault events on an interface, include the `event-thresholds` statement at the `[edit protocols oam ethernet link-fault-management interface]` hierarchy level.

You can also configure OAM threshold values within an action profile and apply the action profile to multiple interfaces. To create an action profile, include the `action-profile` statement at the `[edit protocols oam ethernet link-fault-management]` hierarchy level.

You can configure Ethernet OAM either on an aggregate interface or on each of its member links. However, we recommend that you configure Ethernet OAM on the aggregate interface, and this will internally enable Ethernet OAM on the member links.

To view OAM statistics, use the `show oam ethernet link-fault-management` operational mode command. To clear OAM statistics, use the `clear oam ethernet link-fault-management statistics` operational mode command. To clear link-fault management state information and restart the link discovery process on Ethernet interfaces, use the `clear oam ethernet link-fault-management state` operational mode command. For more information about these commands, see the CLI Explorer.

To enable IEEE 802.3ah OAM support, include the `interface` statement at the `[edit protocols oam ethernet link-fault-management]` hierarchy level:

```
[edit protocols oam ethernet link-fault-management interface interface-name]
```

When you enable IEEE 802.3ah OAM on a physical interface, the discovery process is automatically triggered.

SEE ALSO

> *event-thresholds*
>
> *action-profile*

Release History Table

Release	Description
17.3R1	Starting in Junos OS Release 17.3R1, the Ethernet link fault management daemon (lfmd) runs on the backup Routing Engine as well when graceful Routing Engine switchover (GRES) is configured.

Configure Link Fault Management

IN THIS SECTION

- Configuring Link Discovery | **138**
- Configuring the OAM PDU Interval | **138**
- Configuring the OAM PDU Threshold | **139**
- Configuring Threshold Values for Local Fault Events on an Interface | **139**
- Disabling the Sending of Link Event TLVs | **140**
- Example: Configuring IEEE 802.3ah OAM Support on an Interface | **140**
- Example: Configuring IEEE 802.3ah OAM Support for an Interface on ACX Series | **141**
- Example: Configuring Ethernet LFM Between Provider Edge and Customer Edge | **145**
- Example: Configuring Ethernet LFM for CCC | **146**
- Example: Configuring Ethernet LFM for Aggregated Ethernet | **148**
- Configuring an OAM Action Profile | **151**
- Specifying the Actions to Be Taken for Link-Fault Management Events | **152**
- Monitoring the Loss of Link Adjacency | **153**
- Monitoring Protocol Status | **153**
- Configuring Threshold Values for Fault Events in an Action Profile | **154**
- Applying an Action Profile | **154**

Use this topic to understand how to configure link fault management features on your device. You can also use this topic to configure an action profile to specify the LFM action that must be performed when a specific LFM event occurs and apply the action profile.

Starting in Junos OS Evolved 22.4R1 Release, the Ethernet link fault management process (lfmd) runs only when the `link-fault-management` protocol is configured.

Configuring Link Discovery

When the IEEE 802.3ah OAM protocol is enabled on a physical interface, the discovery process is automatically triggered. The discovery process permits Ethernet interfaces to discover and monitor the peer on the link if it also supports the IEEE 802.3ah standard.

You can specify the discovery mode used for IEEE 802.3ah OAM support. The discovery process is triggered automatically when OAM IEEE 802.3ah functionality is enabled on a port. Link monitoring is done when the interface sends periodic OAM PDUs.

To configure the discovery mode, include the `link-discovery` statement at the `[edit protocol oam ethernet link-fault-management interface` *`interface-name`*`]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name]
link-discovery (active | passive);
```

In active mode, the interface discovers and monitors the peer on the link if the peer also supports IEEE 802.3ah OAM functionality. In passive mode, the peer initiates the discovery process. After the discovery process has been initiated, both sides participate in discovery.

SEE ALSO

> *link-discovery*

Configuring the OAM PDU Interval

Periodic OAM PDUs are sent to perform link monitoring.

You can specify the periodic OAM PDU sending interval for fault detection.

To configure the sending interval, include the `pdu-interval` statement at the `[edit protocol oam ethernet link-fault-management interface` *`interface-name`*`]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name]
pdu-interval interval;
```

The periodic OAM PDU interval range is from 100 through 1000 milliseconds. The default sending interval is 1000 milliseconds.

SEE ALSO

> *pdu-interval*

Configuring the OAM PDU Threshold

You can specify the number of OAM PDUs that an interface can miss before the link between peers is considered down.

To configure the number of PDUs that can be missed from the peer, include the `pdu-threshold` statement at the `[edit protocol oam ethernet link-fault-management interface` *`interface-name`*`]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name]
pdu-threshold threshold-value;
```

The threshold value range is from 3 through 10. The default is three PDUs.

SEE ALSO

> *pdu-threshold*

Configuring Threshold Values for Local Fault Events on an Interface

You can configure threshold values on an interface for the local errors that trigger the sending of link event TLVs.

To set the error threshold values for sending event TLVs, include the `frame-error`, `frame-period`, `frame-period-summary`, and `symbol-period` statements at the `[edit protocols oam ethernet link-fault-management interface` *`interface-name`* `event-thresholds]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name]
event-thresholds {
    frame-error count;
    frame-period count;
    frame-period-summary count;
    symbol-period count;
}
```

SEE ALSO

> *event-thresholds*

frame-error

frame-period

frame-period-summary

symbol-period

Disabling the Sending of Link Event TLVs

You can disable the sending of link event TLVs.

To disable the monitoring and sending of PDUs containing link event TLVs in periodic PDUs, include the `no-allow-link-events` statement at the `[edit protocols oam ethernet link-fault-management interface` *`interface-name`* `negotiation-options]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name negotiation-options]
no-allow-link-events;
```

SEE ALSO

no-allow-link-events

Example: Configuring IEEE 802.3ah OAM Support on an Interface

Configure 802.3ah OAM support on a 10-Gigabit Ethernet interface:

```
[edit]
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface xe-0/0/0 {
                    link-discovery active;
                    pdu-interval 800;
                    pdu-threshold 4;
                    remote-loopback;
                    negotiation-options {
                        allow-remote-loopback;
                    }
                    event-thresholds {
                        frame-error 30;
                        frame-period 50;
```

```
                        frame-period summary 40;
                        symbol-period 20;
                    }
                }
            }
        }
    }
}
```

SEE ALSO

link-fault-management

Example: Configuring IEEE 802.3ah OAM Support for an Interface on ACX Series

IN THIS SECTION

- Requirements | **141**
- Overview and Topology | **142**
- Configuring IEEE 802.3ah OAM on an ACX Series Router | **142**

Junos OS for ACX Series routers allows the Ethernet interfaces on these routers to support the IEEE 802.3ah standard for the Operation, Administration, and Maintenance (OAM) of Ethernet in access networks. The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters.

This example describes how to enable and configure OAM on a Gigabit Ethernet interface.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 12.2 or later for ACX Series routers.
- An ACX1000 or ACX2000 router.

Overview and Topology

In this example, you configure a 10-Gigabit Ethernet interface on an ACX Series router with 802.3ah OAM support, which includes: link discovery, protocol data units (PDUs), remote loopback, negotiation, and event thresholds.

Configuring IEEE 802.3ah OAM on an ACX Series Router

IN THIS SECTION

- CLI Quick Configuration | 142
- Procedure | 142

CLI Quick Configuration

To quickly configure IEEE 802.3ah Ethernet OAM, copy the following commands and paste them into the CLI:

```
edit
edit protocols oam ethernet link-fault-management
set interface xe-0/0/0 link-discovery active pdu-interval 800 pdu-threshold 4 remote-loopback
negotiation-options allow-remote-loopback
set interface xe-0/0/0 event-thresholds frame-error 30 frame-period 50 frame-period-summary 40
symbol-period 20
```

Procedure

Step-by-Step Procedure

To configure IEEE 802.3ah OAM support on an interface:

1. Enable IEEE 802.3ah OAM support on an interface:

   ```
   [edit protocols oam ethernet link-fault-management]
   ```

   user@router1# **set interface (OAM Link-Fault Management) xe-0/0/0**

2. Specify that the interface initiates the discovery process by setting the link discovery mode to **active**:

   user@router# **set interface xe-0/0/0 link-discovery active**

3. Set the periodic OAM PDU-sending interval (in milliseconds) to 800:

   user@router# **set interface xe-0/0/0 pdu-interval 800**

4. Define the number of OAM PDUs to miss before an error is logged as 4:

   user@router# **set interface xe-0/0/0 pdu-threshold 4**

5. Configure the remote interface into loopback mode so that all frames except OAM PDUs are looped back without any changes:

   user@router# **set interface xe-0/0/0 remote-loopback**

6. Configure remote loopback support for the local interface:

   user@router# **set interface xe-0/0/0 negotiation-options allow-remote-loopback**

7. Set the threshold count for sending frame error events to 30:

   user@router# **set interface xe-0/0/0 event-thresholds frame-error 30**

8. Set the threshold count for sending frame period error events to 50:

   user@router# **set interface xe-0/0/0 event-thresholds frame-period 50**

9. Configure the threshold count for sending frame period summary error events to 40:

   user@router# **set interface xe-0/0/0 event-thresholds frame-period-summary 40**

10. Set the threshold count for sending symbol period events to 20:

    user@router# **set interface xe-0/0/0 event-thresholds symbol-period 20**

Results

Check the results of the configuration:

```
[edit]

user@router# show
```

```
[edit]
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface xe-0/0/0 {
                    link-discovery active;
                    pdu-interval 800;
                    pdu-threshold 4;
                    remote-loopback;
                    negotiation-options {
                        allow-remote-loopback;
                    }
                    event-thresholds {
                        frame-error 30;
                        frame-period 50;
                        frame-period-summary 40;
                        symbol-period 20;
                    }
                }
            }
        }
    }
}
```

SEE ALSO

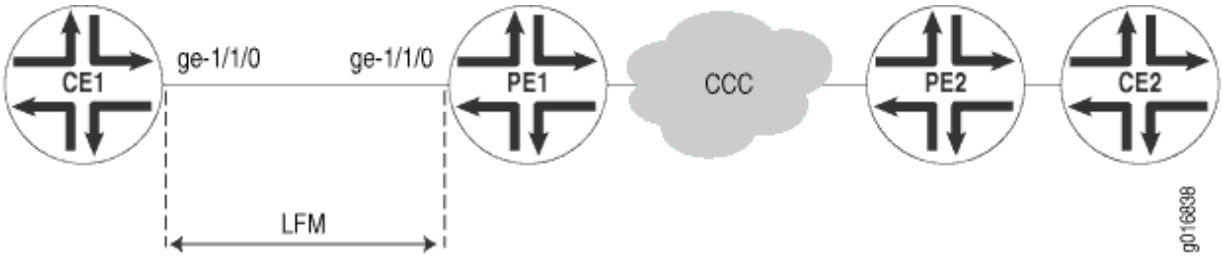
link-fault-management

Example: Configuring Ethernet LFM Between Provider Edge and Customer Edge

In this example, LFM is enabled on an IP link between the provider edge (PE) and customer edge (CE) interfaces. If the link goes down, the fault will be detected by LFM and the interfaces on both sides will be marked **Link-Layer-Down**. This results in notifications to various subsystems (for example, routing) which will take appropriate action.

The link running LFM is shown in Figure 13 on page 145.

Figure 13: Ethernet LFM Between Provider Edge and Customer Edge



To configure Ethernet LFM on an IP link between PE and CE interfaces:

1. Configure LFM on the PE router:

```
[edit]
interfaces ge-1/1/0 {
    unit 0 {
        family inet {
            address 11.11.11.1/24;
        }
    }
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-1/1/0 {
                    pdu-interval 1000;
                    pdu-threshold 5;
                }
            }
        }
```

```
    }
}
```

2. Configure LFM on the CE router:

```
[edit]
interfaces ge-1/1/0 {
    unit 0 {
        family inet {
            address 11.11.11.2/24;
        }
    }
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-1/1/0 {
                    pdu-interval 1000;
                    pdu-threshold 5;
                }
            }
        }
    }
}
```

SEE ALSO

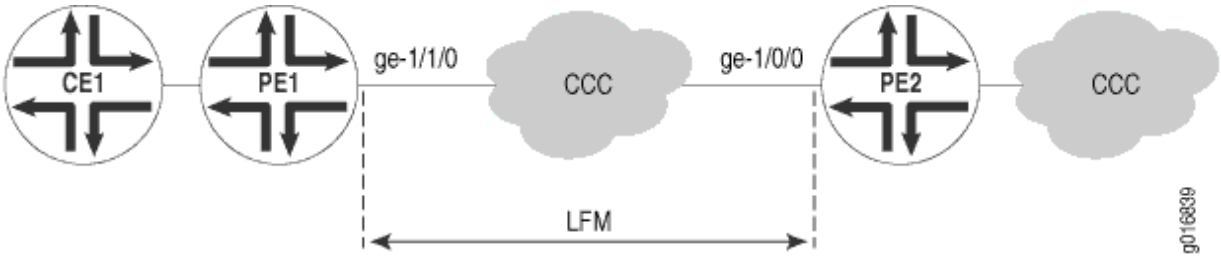
> Ethernet Interfaces User Guide for Routing Devices
>
> IEEE 802.3ah OAM Link-Fault Management Overview | **132**
>
> Example: Configuring Ethernet LFM with Loopback Support | **163**

Example: Configuring Ethernet LFM for CCC

In this example, LFM is configured between two PEs (PE1 and PE2) connected using CCC. With LFM in place, a link fault will be detected immediately, instead of depending on routing protocols to find the fault on end-to-end CCC connection. This also helps in detecting the exact failed link instead of only finding that the end-to-end CCC connectivity has failed. Also, because LFM runs at the link-layer level, it does not need a IP address to operate and so can be used where bidirectional fault detection (BFD) cannot.

The links running LFM are shown in Figure 14 on page 147

Figure 14: Ethernet LFM for CCC



To configure Ethernet LFM between two PEs connected using CCC:

1. Configure LFM on the PE1 router with CCC:

```
[edit]
interfaces ge-1/1/0 {
    encapsulation ethernet-ccc;
    unit 0;
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-1/1/0 {
                    pdu-interval 1000;
                    pdu-threshold 5;
                }
            }
        }
    }
}
```

2. Configure LFM on the PE2 router with CCC:

```
[edit]
interfaces ge-1/0/0 {
    encapsulation ethernet-ccc;
    unit 0;
}
```

```
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-1/0/0 {
                    pdu-interval 1000;
                    pdu-threshold 5;
                }
            }
        }
    }
}
```

SEE ALSO

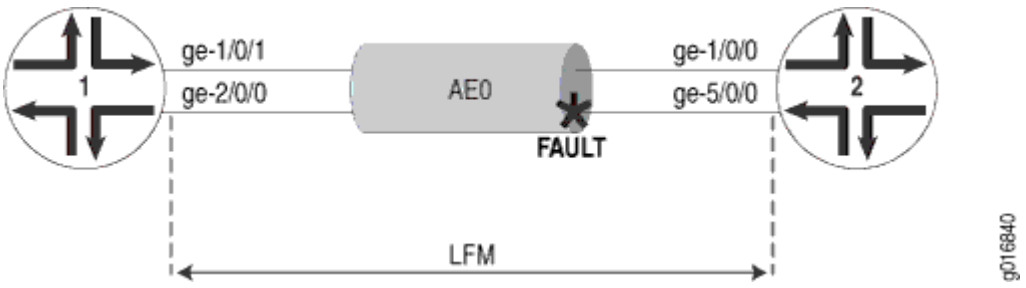
- Ethernet Interfaces User Guide for Routing Devices
- IEEE 802.3ah OAM Link-Fault Management Overview | **132**
- Example: Configuring Ethernet LFM with Loopback Support | **163**

Example: Configuring Ethernet LFM for Aggregated Ethernet

In this example, LFM is configured on an aggregated Ethernet interface (AE0) between Router 1 and Router 2. When configured on aggregated Ethernet, LFM runs on all the individual member links. LFM is enabled or disabled on the member links as they are added or deleted from the aggregation group. The status of individual links is used to determine the status of the aggregated interface.

The use of LFM with aggregated Ethernet is shown in Figure 15 on page 148.

Figure 15: Ethernet LFM for Aggregated Ethernet



To configure LFM on an aggregated Ethernet interface between two routers:

1. Configure LFM on Router 1 for AE0:

```
[edit]
chassis {
    aggregated-devices {
        ethernet {
            device-count 1;
        }
    }
}
interfaces ge-1/0/1 {
    gigether-options {
        802.3ad ae0;
    }
}
interfaces ge-2/0/0 {
    gigether-options {
        802.3ad ae0;
    }
}
interfaces ae0 {
    unit 0 {
        family inet {
            address 11.11.11.2/24;
        }
    }
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ae0;
            }
        }
    }
}
```

2. Configure LFM on Router 2 for AE0:

```
[edit]
chassis {
```

```
    aggregated-devices {
        ethernet {
            device-count 1;
        }
    }
}
interfaces ge-1/0/0 {
    gigether-options {
        802.3ad ae0;
    }
}
interfaces ge-5/0/0 {
    gigether-options {
        802.3ad ae0;
    }
}
interfaces ae0 {
    unit 0 {
        family inet {
            address 11.11.11.1/24;
        }
    }
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ae0;
            }
        }
    }
}
```

SEE ALSO

Ethernet Interfaces User Guide for Routing Devices

IEEE 802.3ah OAM Link-Fault Management Overview | **132**

Example: Configuring Ethernet LFM with Loopback Support | **163**

Configuring an OAM Action Profile

You can create an action profile to define event fault flags and thresholds and the action to be taken. You can then apply the action profile to one or more interfaces.

To configure an action profile, include the `action-profile` statement at the `[edit protocols oam ethernet link-fault-management]` hierarchy level:

```
action-profile profile-name {
    action {
        syslog;
        link-down;
        send-critical-event;
    }
    event {
        link-adjacency-loss;
        link-event-rate {
            frame-error count;
            frame-period count;
            frame-period-summary count;
            symbol-period count;
        }
        protocol-down;
    }
}
```

NOTE: Starting from Junos OS Release 14.2, whenever link-fault management (LFM) with an action profile is configured to mark the interface as down (by including the link-down statement at the [edit protocols oam ethernet link-fault-management] hierarchy level), the port is placed in the blocked state (STP state). In such a state of the interface, data traffic is not transmitted out on that interface. Because the connectivity-fault management (CFM) downstream maintenance MEPs come up on blocked ports, the CFM sessions come up properly. However, the interface is down and the interface status TLV does not contain the correct status. Only if you configure the port status TLV, the actual status of the port is reflected. The interface status TLV does not carry the actual state of the port.

SEE ALSO

Setting a Remote Interface into Loopback Mode | 158

Enabling Remote Loopback Support on the Local Interface | 159

Specifying the Actions to Be Taken for Link-Fault Management Events

You can specify the action to be taken by the system when the configured link-fault event occurs. Multiple action profiles can be applied to a single interface. For each action-profile, at least one event and one action must be specified. The actions are taken only when all of the events in the action profile are true. If more than one action is specified, all the actions are executed.

You might want to set a lower threshold for a specific action such as logging the error and set a higher threshold for another action such as sending a critical event TLV.

To specify the action, include the `action` statement at the `[edit protocols oam ethernet link-fault-management action-profile` *`profile-name`*`]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management action-profile profile-name]
event {
    link-adjacency-loss;
    protocol-down;
}
action {
    syslog;
    link-down;
    send-critical-event;
}
```

To create a system log entry when the link-fault event occurs, include the `syslog` statement.

To administratively disable the link when the link-fault event occurs, include the `link-down` statement.

To send IEEE 802.3ah link event TLVs in the OAM PDU when a link-fault event occurs, include the `send-critical-event` statement.

NOTE: If multiple actions are specified in the action profile, all of the actions are executed in no particular order.

SEE ALSO

action

syslog

link-down

send-critical-event

Monitoring the Loss of Link Adjacency

You can specify actions be taken when link adjacency is lost. When link adjacency is lost, the system takes the action defined in the `action` statement of the action profile.

To configure the system to take action when link adjacency is lost, include the `link-adjacency-loss` statement at the `[edit protocols oam ethernet link-fault-management action-profile` *`profile-name`* `event]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management action-profile profile-name]
link-adjacency-loss;
```

SEE ALSO

link-adjacency-loss

Enabling Remote Loopback Support on the Local Interface | 159

Monitoring Protocol Status

The CCC-DOWN flag is associated with a circuit cross-connect (CCC) connection, Layer 2 circuit, and Layer 2 VPN, which send the CCC-DOWN status to the kernel. The CCC-DOWN flag indicates that the CCC is down. The CCC-DOWN status is sent to the kernel when the CCC connection, Layer 2 circuit, or Layer 2 VPN is down. This in turn, brings down the CE-facing PE interface associated with the CCC connection, Layer 2 circuit, or Layer 2 VPN.

When the CCC-DOWN flag is signaled to the IEEE 802.3ah protocol, the system takes the action defined in the `action` statement of the action profile. For additional information about Layer 2 circuits, see the Junos OS Layer 2 Circuits User Guide, Junos OS VPNs Configuration Guide.

To monitor the IEEE 802.3ah protocol, on the CE-facing PE interface, include the `protocol-down` statement at the `[edit protocols oam ethernet link-fault-management action-profile` *`profile-name`* `event]` hierarchy level:

1. In configuration mode, go to the `[edit protocols oam ethernet link-fault-management action-profile profile-name event]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet link-fault-management action-profile profile-name event
```

2. Include the `protocol-down` statement.

```
[edit protocols oam ethernet link-fault-management action-profile profile-name event]
user@host# set protocol-down
```

NOTE: If multiple events are specified in the action profile, all the events must occur before the specified action is taken.

SEE ALSO

protocol-down

Setting a Remote Interface into Loopback Mode | 158

Enabling Remote Loopback Support on the Local Interface | 159

Configuring Threshold Values for Fault Events in an Action Profile

You can configure link event thresholds for received error events that trigger the action specified in the `action` statement. You can then apply the action profile to one or more interfaces.

To configure link event thresholds, include the `link-event-rate` statement at the `[edit protocols oam ethernet link-fault-management action-profile` *`profile-name`* `event]` hierarchy level:

```
link-event-rate {
    frame-error count;
    frame-period count;
    frame-period-summary count;
    symbol-period count;
}
```

SEE ALSO

link-event-rate

Applying an Action Profile

You can apply an action profile to one or more interfaces.

To apply an action profile to an interface, include the `apply-action-profile` statement at the `[edit protocols oam ethernet link-fault-management action-profile interface interface-name]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name]
apply-action-profile profile-name;
```

SEE ALSO

apply-action-profile

Release History Table

Release	Description
14.2	Starting from Junos OS Release 14.2

RELATED DOCUMENTATION

Remote Loopback for Link Fault Management | 158

Remote Fault Detection for Link Fault Management

IN THIS SECTION

- Detecting Remote Faults | 155
- Enabling Dying Gasp Functionality | 156

Use this topic to understand more about remote faults and how they are detected and also how to enable the dying gasp feature to avoid file system corruption for LFM.

Detecting Remote Faults

Fault detection is either based on flags or fault event type, length, and values (TLVs) received in OAM protocol data units (PDUs). Flags that trigger a link fault are:

- Critical Event
- Dying Gasp
- Link Fault

The link event TLVs are sent by the remote DTE by means of event notification PDUs. Link event TLVs are:

- Errored Symbol Period Event
- Errored Frame Event
- Errored Frame Period Event
- Errored Frame Seconds Summary Event

SEE ALSO

> IEEE 802.3ah OAM Link-Fault Management Overview | **132**
>
> Configuring IEEE 802.3ah OAM Link-Fault Management

Enabling Dying Gasp Functionality

Dying gasp means an unrecoverable condition such as a power failure. In this condition, the local peer informs the remote peer about the failure state. When the remote peer receives a dying-gasp PDU, it takes an action corresponding to the action profile configured with the **link-adjacency-loss** event. Dying gasp helps to avoid file system corruption.

NOTE: ACX5096 and ACX5048 routers do not support dying-gasp.

ACX Series routers can generate and receive dying-gasp packets. When LFM is configured on an interface, a dying-gasp PDU is generated for the interface on the following failure conditions:

- Power failure
- Packet Forwarding Engine panic or a crash

ACX Series routers support the following CLI statements to enable dying-gasp functionality:

- `dgasp-int`—Enables dying-gasp functionality.
- `dgasp-usb`—Resets USB port during dying-gasp event.

The dgasp-int and dgasp-usb CLI statements are added under the [edit system] hierarchy to enable dying-gasp functionality.

To enable dying-gasp functionality, you need to configure the dgasp-int and dgasp-usb CLI statements as shown below:

```
root@host% cli
root@host> configure
Entering configuration mode

[edit]
root@host# set system dgasp-int

[edit]
root@host# set system dgasp-usb

[edit]
root@host# commit

commit complete

[edit]
root@host# show system
dgasp-int;
dgasp-usb;
```

The dying-gasp functionality is disabled by default.

SEE ALSO

Understanding Ethernet OAM Link Fault Management for ACX Series Routers | 133

RELATED DOCUMENTATION

Introduction to OAM Link Fault Management (LFM) | 132

Remote Loopback for Link Fault Management

IN THIS SECTION

- Setting a Remote Interface into Loopback Mode | 158
- Enabling Remote Loopback Support on the Local Interface | 159
- Enabling Nonstop Routing for Ethernet Link Fault Management on Backup Routers | 159
- Example: Configuring Ethernet LFM with Loopback Support | 163

Use this topic to understand what happens when you set a remote interfaces in loopback mode and how to enable remote loopback. You can also learn how to enable nonstop routing for LFM.

Setting a Remote Interface into Loopback Mode

You can configure the software to set the remote DTE into loopback mode on the following interfaces:

- IQ2 and IQ2-E Gigabit Ethernet interfaces
- Ethernet interfaces on the MX Series routers or EX Series switches

Junos OS can place a remote DTE into loopback mode (if remote-loopback mode is supported by the remote DTE). When you place a remote DTE into loopback mode, the interface receives the remote-loopback request and puts the interface into remote-loopback mode. When the interface is in remote-loopback mode, all frames except OAM PDUs are looped back without any changes made to the frames. OAM PDUs continue to be sent to the management plane and processed.

To configure remote loopback, include the `remote-loopback` statement at the `[edit protocol oam ethernet link-fault-management interface` *`interface-name`*`]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name]
remote-loopback;
```

To take the remote DTE out of loopback mode, remove the `remote-loopback` statement from the configuration.

SEE ALSO

remote-loopback

Enabling Remote Loopback Support on the Local Interface

You can allow a remote DTE to set a local interface into remote loopback mode on IQ2 and IQ2-E Gigabit Ethernet interfaces and all Ethernet interfaces on the MX Series routers and EX Series switches. When a remote-loopback request is sent by a remote DTE, the Junos OS places the local interface into loopback mode. When an interface is in loopback mode, all frames except OAM PDUs are looped back without any changes to the frames. OAM PDUs continue to be sent to the management plane and processed. By default, the remote loopback feature is not enabled.

To enable remote loopback, include the `allow-remote-loopback` statement at the `[edit protocol oam ethernet link-fault-management interface` *`interface-name`* `negotiation-options]` hierarchy level:

```
[edit protocol oam ethernet link-fault-management interface interface-name negotiation-options]
allow-remote-loopback;
```

NOTE: Activation of OAM remote loopback may result in data frame loss.

SEE ALSO

allow-remote-loopback

Enabling Nonstop Routing for Ethernet Link Fault Management on Backup Routers

Starting in Junos OS Release 17.3R1, the Ethernet link fault management daemon (lfmd) runs on the backup Routing Engine as well when graceful Routing Engine switchover (GRES) is configured. When the lfmd daemon runs on the backup Routing Engine as well, the link fault management states are kept in sync and so minimal effort is required by the lfmd daemon post switch over.

To enable Nonstop routing for Ethernet LFM on backup routers:

1. Enable graceful Routing Engine switchover. By default, GRES is disabled. To enable GRES, include the `graceful-switchover` statement at the `[edit chassis redundancy]` hierarchy level. By default, Nonstop routing is disabled. When you enable GRES, NSR is enabled.

   ```
   [edit chassis redundancy]
   user@host# set graceful-switchover
   ```

2. Synchronize the Routing Engine configuration. To synchronize the primary Routing Engine configuration with the backup, include the `synchronize` statement at the [`edit system`] hierarchy level.

```
[edit system]
user@host# set commit synchronize
```

3. After enabling nonstop routing, commit the configuration.

```
[edit routing options]
user@host# commit
```

4. To verify if nonstop routing is enabled on the backup router, at the operational mode, use the `show oam ethernet link-fault-management` command on the primary router and then the backup router. Because you have enabled synchronization, the output of the primary router and the backup router is identical. However, the statistics maintained by the primary router are not synchronized with the backup router..

```
{master}
user@host# show oam ethernet link-fault-management ge-0/2/0 detail
```

```
Interface: ge-0/2/0
    Status: Running, Discovery state: Send Any
    Transmit interval: 100ms, PDU threshold: 3 frames, Hold time: 300ms
    Peer address: ac:4b:c8:81:90:a4
    Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
    OAM receive statistics:
      Information: 0, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 0, Organization specific: 0
    OAM flags receive statistics:
      Critical event: 0, Dying gasp: 0, Link fault: 0
    OAM transmit statistics:
      Information: 0, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 786, Organization specific: 0
    OAM received symbol error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame period error event information:
```

```
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame seconds error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM transmitted symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM current symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM transmitted frame error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM current frame error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    Loopback tracking: Enabled, Loop status: Not Found
    Detect LOC: Enabled, LOC status: Not Found
    Remote entity information:
      Remote MUX action: forwarding, Remote parser action: forwarding
      Discovery mode: active, Unidirectional mode: unsupported
      Remote loopback mode: unsupported, Link events: supported
      Variable requests: unsupported
Application profile statistics:
  Profile Name                   Invoked     Executed
  LK_ADJ_LOSS100_1                     1            1
  LK_ADJ_LOSS100_2                     1            0
  LK_ADJ_LOSS100_3                     1            0
  LK_ADJ_LOSS101_1                     1            1
  LK_ADJ_LOSS101_2                     1            0
  LK_ADJ_LOSS101_3                     1            0
  LK_ADJ_LOSS106_1                     0            0
  LK_ADJ_LOSS106_2                     0            0
  LK_ADJ_LOSS106_3                     0            0
  LK_ADJ_LOSS107_1                     0            0
```

```
  LK_ADJ_LOSS107_2                          0        0
  LK_ADJ_LOSS107_3                          0        0
```

```
{backup}
user@host# show oam ethernet link-fault-management ge-0/2/0 detail
```

```
Interface: ge-0/2/0
    Status: Running, Discovery state: Send Any
    Transmit interval: 100ms, PDU threshold: 3 frames, Hold time: 300ms
    Peer address: ac:4b:c8:81:90:a4
    Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
    OAM receive statistics:
      Information: 0, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 0, Organization specific: 0
    OAM flags receive statistics:
      Critical event: 0, Dying gasp: 0, Link fault: 0
    OAM transmit statistics:
      Information: 0, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 786, Organization specific: 0
    OAM received symbol error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame period error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame seconds error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM transmitted symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM current symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM transmitted frame error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
```

```
    OAM current frame error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    Loopback tracking: Enabled, Loop status: Not Found
    Detect LOC: Enabled, LOC status: Not Found
    Remote entity information:
      Remote MUX action: forwarding, Remote parser action: forwarding
      Discovery mode: active, Unidirectional mode: unsupported
      Remote loopback mode: unsupported, Link events: supported
      Variable requests: unsupported
Application profile statistics:
  Profile Name                  Invoked     Executed
  LK_ADJ_LOSS100_1                    0            0
  LK_ADJ_LOSS100_2                    0            0
  LK_ADJ_LOSS100_3                    0            0
  LK_ADJ_LOSS101_1                    0            0
  LK_ADJ_LOSS101_2                    0            0
  LK_ADJ_LOSS101_3                    0            0
  LK_ADJ_LOSS106_1                    0            0
  LK_ADJ_LOSS106_2                    0            0
  LK_ADJ_LOSS106_3                    0            0
  LK_ADJ_LOSS107_1                    0            0
  LK_ADJ_LOSS107_2                    0            0
  LK_ADJ_LOSS107_3                    0            0
```

NOTE: After the switchover, if issues are observed, use the `clear oam ethernet link-fault-management state` command for specific sessions. If the issue does not get resolved, restart the lfmd daemon.

SEE ALSO

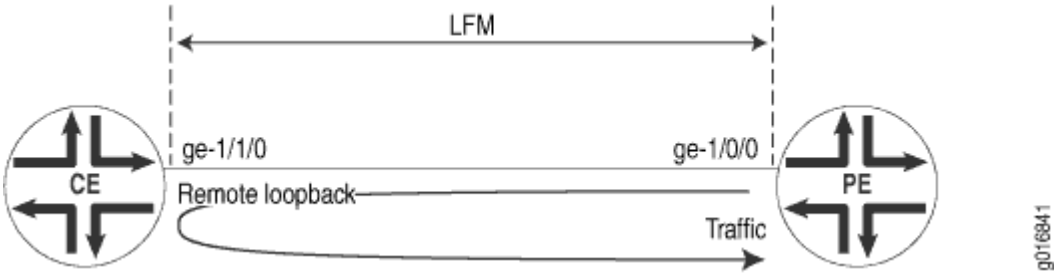
IEEE 802.3ah OAM Link-Fault Management Overview | 132

show oam ethernet link-fault-management

Example: Configuring Ethernet LFM with Loopback Support

In this example, LFM is configured between provider edge (PE) router and the customer edge (CE) router. The PE router can put the CE router in remote loopback mode. This allows the PE to have all the traffic sent to the CE router looped back for diagnostics purposes, as shown in Figure 16 on page 164.

Figure 16: Ethernet LFM with Loopback Support



To configure LFM between a PE router and a CE router:

1. Configure LFM loopback on the PE router:

```
[edit]
interfaces ge-1/0/0 {
    unit 0 {
        family inet {
            address 11.11.11.1/24;
        }
    }
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-1/0/0 {
                    pdu-interval 1000;
                    pdu-threshold 5;
                    remote-loopback;
                }
            }
        }
    }
}
```

2. Configure LFM loopback on the CE router:

```
[edit]
interfaces ge-1/1/0 {
    unit 0 {
        family inet {
```

```
            address 11.11.11.2/24;
        }
    }
}
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-1/1/0 {
                    pdu-interval 1000;
                    pdu-threshold 5;
                    negotiation-options {
                        allow-remote-loopback;
                    }
                }
            }
        }
    }
}
```

NOTE: If the `negotiation options allow-remote-loopback` statement on the CE router is deleted before removing the CE router from remote loopback mode, traffic flow between the PE router and CE router is affected. Hence, delete the `remote-loopback` statement on the PE router before deleting the `negotiation-options allow-remote-loopback` statement on the CE router.

SEE ALSO

Example: Configuring Ethernet LFM Between Provider Edge and Customer Edge | 145

Example: Configuring Ethernet LFM for CCC | 146

Example: Configuring Ethernet LFM for Aggregated Ethernet | 148

Release History Table

Release	Description
17.3R1	Starting in Junos OS Release 17.3R1, the Ethernet link fault management daemon (lfmd) runs on the backup Routing Engine as well when graceful Routing Engine switchover (GRES) is configured.

RELATED DOCUMENTATION

Introduction to OAM Link Fault Management (LFM) | 132

Configure Link Fault Management | 137

CHAPTER 3

Ethernet OAM Link Fault Management for Switches

IN THIS CHAPTER

- Ethernet OAM Link Fault Management | 167
- Configure Ethernet OAM Link Fault Management | 168
- Example: Configure Ethernet OAM Link Fault Management | 172

Ethernet OAM Link Fault Management

Juniper Networks Junos operating system (Junos OS) for Juniper Networks allows the Ethernet interfaces on these switches to support the IEEE 802.3ah standard for the Operation, Administration, and Maintenance (OAM) of Ethernet in access networks. The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters. The IEEE 802.3ah standard meets the requirement for OAM capabilities even as Ethernet moves from being solely an enterprise technology to a WAN and access technology, and the standard remains backward-compatible with existing Ethernet technology.

Ethernet OAM provides the tools that network management software and network managers can use to determine how a network of Ethernet links is functioning. Ethernet OAM should:

- Rely only on the media access control (MAC) address or virtual LAN identifier for troubleshooting.
- Work independently of the actual Ethernet transport and function over physical Ethernet ports or a virtual service such as pseudowire.
- Isolate faults over a flat (or single operator) network architecture or nested or hierarchical (or multiprovider) networks.

The following OAM LFM features are supported:

- Discovery and Link Monitoring

  The discovery process is triggered automatically when OAM is enabled on the interface. The discovery process permits Ethernet interfaces to discover and monitor the peer on the link if it also supports the IEEE 802.3ah standard. You can specify the discovery mode used for IEEE 802.3ah

OAM support. In active mode, the interface discovers and monitors the peer on the link if the peer also supports IEEE 802.3ah OAM functionality. In passive mode, the peer initiates the discovery process. After the discovery process has been initiated, both sides participate in discovery. The switch performs link monitoring by sending periodic OAM protocol data units (PDUs) to advertise OAM mode, configuration, and capabilities.

You can specify the number of OAM PDUs that an interface can miss before the link between peers is considered down.

- Remote Fault Detection

  Remote fault detection uses flags and events. Flags are used to convey the following: Link Fault means a loss of signal, Dying Gasp means an unrecoverable condition such as a power failure, and Critical Event means an unspecified vendor-specific critical event. You can specify the periodic OAM PDU sending interval for fault detection. The switch uses the Event Notification OAM PDU to notify the remote OAM device when a problem is detected. You can specify the action to be taken by the system when the configured link-fault event occurs.

- Remote Loopback Mode

  Remote loopback mode ensures link quality between the switch and a remote peer during installation or troubleshooting. In this mode, when the interface receives a frame that is not an OAM PDU or a pause frame, it sends it back on the same interface on which it was received. The link appears to be in the active state. You can use the returned loopback acknowledgement to test delay, *jitter*, and throughput.

  Junos OS can place a remote DTE into loopback mode (if remote loopback mode is supported by the remote DTE). When you place a remote DTE into loopback mode, the interface receives the remote loopback request and puts the interface into remote loopback mode. When the interface is in remote loopback mode, all frames except OAM PDUs are looped back without any changes made to the frames. OAM PDUs continue to be sent and processed.

Configure Ethernet OAM Link Fault Management

Ethernet OAM link fault management (LFM) can be used for physical link-level fault detection and management. The IEEE 802.3ah LFM works across point-to-point Ethernet links either directly or through repeaters.

To configure Ethernet OAM LFM using the CLI:

1. Enable IEEE 802.3ah OAM support on an interface:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface                          interface-
name
```

NOTE: You can configure Ethernet OAM LFM on aggregated interfaces.

NOTE: The remaining steps are optional. You can choose which of these features to configure for Ethernet OAM LFM on your switch.

2. Specify whether the interface or the peer initiates the discovery process by configuring the link discovery mode to `active` or `passive` (`active` = interface initiates; `passive` = peer initiates):

```
[edit protocols oam ethernet link-fault-management]
 user@switch# set interface interface-namelink-discovery active
```

3. Configure a periodic OAM PDU-sending interval (in milliseconds) for fault detection:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface pdu-intervalinterval
```

4. Specify the number of OAM PDUs that an interface can miss before the link between peers is considered down:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name                          pdu-
threshold                          threshold-value
```

5. Configure event threshold values on an interface for the local errors that trigger the sending of link event TLVs:

   - Set the threshold value (in seconds) for sending frame-error events or taking the action specified in the action profile:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name                                  event-
```

```
thresholds                                    frame-error
count
```

- Set the threshold value (in seconds) for sending frame-period events or taking the action specified in the action profile:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name event-thresholds frame-
period
count
```

- Set the threshold value (in seconds) for sending frame-period-summary events or taking the action specified in the action profile:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name event-thresholds frame-period-
summary
count
```

- Set the threshold value (in seconds) for sending symbol-period events or taking the action specified in the action profile:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name event-thresholds symbol-
period
count
```

NOTE: You can disable the sending of link event TLVs.

To disable the sending of link event TLVs:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name                          negotiation-
options                          no-allow-link-events
```

6. Create an action profile to define event fault flags and thresholds to be taken when the link fault event occurs. Then apply the action profile to one or more interfaces. (You can also apply multiple action profiles to a single interface.)

   a. Name the action profile:

```
[edit protocols oam ethernet link-fault-management]
user@switch# set action-profile                                    profile-
name
```

   b. Specify actions to be taken by the system when the link fault event occurs:

```
[edit protocols oam ethernet link-fault-management]
 user@switch# set action-profile profile-name
action
syslog
```

```
user@switch# set action-profile profile-name action link-
down
```

   c. Specify events for the action profile:

```
[edit protocols oam ethernet link-fault-management]
 user@switch# set action-profile profile-name event link-adjacency-
loss
```

NOTE: For each action profile, you must specify at least one link event and one action. The actions are taken only when all of the events in the action profile are true. If more than one action is specified, all actions are executed. You can set a low threshold for a specific action such as logging the error and set a high threshold for another action such as system logging.

7. Set a remote interface into loopback mode so that all frames except OAM PDUs are looped back without any changes made to the frames. Set the remote DTE in loopback mode (the remote DTE

must support remote-loopback mode) and then enable remote loopback support for the local interface.

```
[edit protocols oam ethernet link-fault-management]
user@switch# set interface interface-name                    remote-
loopback
```

```
user@switch# set interface interface-name negotiation-options allow-remote-
loopback
```

Example: Configure Ethernet OAM Link Fault Management

IN THIS SECTION

- Requirements | **172**
- Overview and Topology | **173**
- Configuring Ethernet OAM Link Fault Management on Switch 1 | **173**
- Configuring Ethernet OAM Link Fault Management on Switch 2 | **175**
- Verification | **177**

Junos OS allows the Ethernet interfaces on these switches to support the IEEE 802.3ah standard for the Operation, Administration, and Maintenance (OAM) of Ethernet in access networks. The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters.

This example describes how to enable and configure OAM LFM on a Gigabit Ethernet interface:

Requirements

This example uses the following hardware and software components:

- Junos OS Release 9.4 or later for EX Series switches
- Two EX3200 or EX4200 switches connected directly

Overview and Topology

IN THIS SECTION

- Topology | 173

Junos OS switches allows the Ethernet interfaces on these switches to support the IEEE 802.3ah standard for the Operation, Administration, and Maintenance (OAM) of Ethernet in access networks. The standard defines OAM link fault management (LFM). You can configure IEEE 802.3ah OAM LFM on point-to-point Ethernet links that are connected either directly or through Ethernet repeaters.

Topology

This example uses two EX4200 switches connected directly. Before you begin configuring Ethernet OAM LFM on two switches, connect the two switches directly through a trunk interface.

Configuring Ethernet OAM Link Fault Management on Switch 1

IN THIS SECTION

- CLI Quick Configuration | 173
- Procedure | 174
- Results | 175

CLI Quick Configuration

To quickly configure Ethernet OAM LFM, copy the following commands and paste them into the switch terminal window:

```
[edit protocols oam ethernet link-fault-management]
                    set interface ge-0/0/0
                    set interface ge-0/0/0 link-discovery active
                    set interface ge-0/0/0 pdu-interval 800
                    set interface ge-0/0/0 remote-loopback
```

Procedure

Step-by-Step Procedure

To configure Ethernet OAM LFM on switch 1:

1. Enable IEEE 802.3ah OAM support on an interface:

```
[edit protocols oam ethernet link-fault-management]
user@switch1# set interface ge-0/0/0
```

2. Specify that the interface initiates the discovery process by configuring the link discovery mode to `active`:

```
[edit protocols oam ethernet link-fault-management]
user@switch1# set interface ge-0/0/0 link-discovery active
```

3. Set the periodic OAM PDU-sending interval (in milliseconds) to 800 on switch 1:

```
[edit protocols oam ethernet link-fault-management]
user@switch1# set interface pdu-interval 800
```

4. Set a remote interface into loopback mode so that all frames except OAM PDUs are looped back without any changes made to the frames. Ensure that the remote DTE supports remote loopback mode. To set the remote DTE in loopback mode

```
[edit protocols oam ethernet link-fault-management]
user@switch1# set interface ge-0/0/0.0 remote-
loopback
```

Results

Check the results of the configuration:

```
[edit]

user@switch1# show
```

```
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-0/0/0 {
                    pdu-interval 800;
                    link-discovery active;
                    remote-loopback;
                }
            }
        }
    }
```

Configuring Ethernet OAM Link Fault Management on Switch 2

IN THIS SECTION

- CLI Quick Configuration | 175
- Procedure | 176

CLI Quick Configuration

To quickly configure Ethernet OAM LFM on switch 2, copy the following commands and paste them into the switch terminal window:

```
[edit protocols oam ethernet link-fault-management ]
                    set interface ge-0/0/1
```

```
set interface ge-0/0/1 negotiation-options allow-remote-loopback
```

Procedure

Step-by-Step Procedure

To configure Ethernet OAM LFM on switch 2:

1. Enable OAM on the peer interface on switch 2:

```
[edit protocols oam ethernet link-fault-management]
user@switch2# set interface ge-0/0/1
```

2. Enable remote loopback support for the local interface:

```
[edit protocols oam ethernet link-fault-management]
user@switch2# set interface ge-0/0/1 negotiation-options allow-remote-
loopback
```

Results

Check the results of the configuration:

```
[edit]
user@switch2# show
```

```
protocols {
    oam {
        ethernet {
            link-fault-management {
                interface ge-0/0/1 {
                    negotiation-options {
                        allow-remote-loopback;
                    }
                }
            }
```

```
        }
    }
```

Verification

IN THIS SECTION

- Verifying That OAM LFM Has Been Configured Properly | 177

Verifying That OAM LFM Has Been Configured Properly

Purpose

Verify that OAM LFM has been configured properly.

Action

Use the `show oam ethernet link-fault-management` command:

```
user@switch1#
```

Sample Output

command-name

```
Interface: ge-0/0/0.0
Status: Running, Discovery state: Send Any
Peer address: 00:19:e2:50:3b:e1
Flags:Remote-Stable Remote-State-Valid Local-Stable 0x50
Remote entity information:
Remote MUX action: forwarding, Remote parser action: forwarding
Discovery mode: active, Unidirectional mode: unsupported
Remote loopback mode: supported, Link events: supported
Variable requests: unsupported
```

Meaning

When the output displays the MAC address and the discover state is `Send Any`, it means that OAM LFM has been configured properly.

CHAPTER 4

Ethernet OAM Connectivity Fault Management for Switches

IN THIS CHAPTER

- Understand Ethernet OAM Connectivity Fault Management for Switches | **179**
- Configure Ethernet OAM Connectivity Fault Management (CLI Procedure) | **182**
- Example: Configure Ethernet OAM Connectivity Fault Management on EX Series Switches | **188**

Understand Ethernet OAM Connectivity Fault Management for Switches

IN THIS SECTION

- CFM Limitations on EX4600 Switches | **181**
- CFM Limitations on QFX5120, QFX5200, and QFX5210 Series Switches | **181**

The IEEE 802.1ag specification provides for Ethernet connectivity fault management (CFM). CFM monitors Ethernet networks that might comprise one or more service instances for network-compromising connectivity faults.

The major features of CFM are:

- Fault monitoring using the continuity check protocol. This is a neighbor discovery and health check protocol that discovers and maintains adjacencies at the VLAN level.
- Path discovery and fault verification using the linktrace protocol.
- Fault isolation using the loopback protocol.

CFM partitions the service network into various administrative domains. For example, operators, providers, and customers might be part of different administrative domains. Each administrative domain

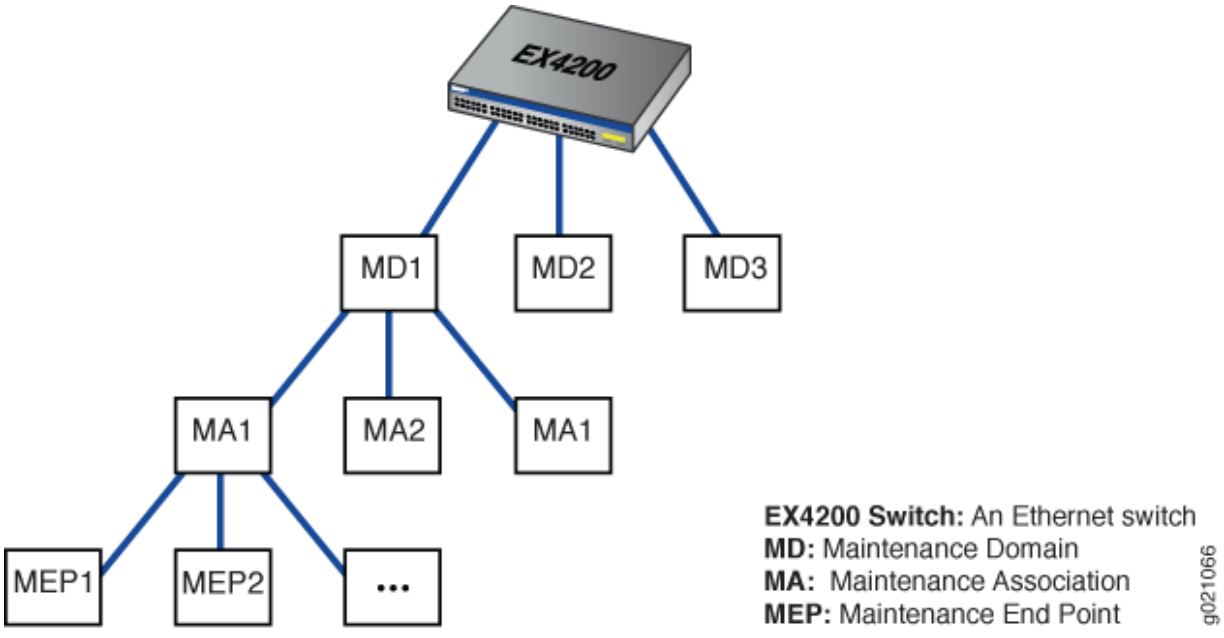
is mapped into one maintenance domain providing enough information to perform its own management, thus avoiding security breaches and making end-to-end monitoring possible.

In a CFM maintenance domain, each service instance is called a maintenance association. A maintenance association can be thought of as a full mesh of maintenance association endpoints (MEPs) having similar characteristics. MEPs are active CFM entities generating and responding to CFM protocol messages. There is also a maintenance intermediate point (MIP), which is a CFM entity similar to the MEP, but more passive (MIPs only respond to CFM messages).

Each maintenance domain is associated with a maintenance domain level from 0 through 7. Level allocation is based on the network hierarchy, where outer domains are assigned a higher level than the inner domains. Configure customer end points to have the highest maintenance domain level. The maintenance domain level is a mandatory parameter that indicates the nesting relationships between various maintenance domains. The level is embedded in each CFM frame. CFM messages within a given level are processed by MEPs at that same level.

To enable CFM on an Ethernet interface, you must configure maintenance domains, maintenance associations, and maintenance association end points (MEPs). Figure 17 on page 180 shows the relationships among maintenance domains, maintenance association end points (MEPs), and maintenance intermediate points (MIPs) configured on a switch.

Figure 17: Relationship Among MEPs, MIPs, and Maintenance Domain Levels



CFM Limitations on EX4600 Switches

Starting in Junos OS Release 18.3R1, Junos OS provides CFM support on EX4600. CFM support on EX4600 has the following limitations:

- CFM support is provided via software using filters. This can impact scaling.
- Inline Packet Forwarding Engine (PFE) mode is not supported. In Inline PFE mode, you can delegate periodic packet management (PPM) processing to the Packet Forwarding Engine (PFE) which results in faster packet handling and the CCM interval supported is 10 milliseconds.
- Performance monitoring (ITU-T Y.1731 Ethernet Service OAM) is not supported.
- CCM interval of less than 1 second is not supported.
- CFM is not supported on Routed Interfaces and aggregated Ethernet (lag) interfaces.
- MIP half function, to divide the MIP functionality into two unidirectional segments to improve network coverage, is not supported.
- Up MEP is not supported.
- Total number of CFM sessions supported is 20.

CFM Limitations on QFX5120, QFX5200, and QFX5210 Series Switches

Starting in Junos OS Release 18.4R1, Junos OS provides CFM support on QFX5200 switches and QFX5210 switches. Starting in Junos OS Release 19.4R1, Junos OS provides CFM support on QFX5120 switches. CFM support on QFX5120, QFX5200, and QFX5210 Series switches has the following limitations:

- CFM support is provided via software using filters. This can impact scaling.
- Inline Packet Forwarding Engine (PFE) mode is not supported. In Inline PFE mode, you can delegate periodic packet management (PPM) processing to the Packet Forwarding Engine (PFE) which results in faster packet handling and the CCM interval supported is 10 milliseconds.
- Performance monitoring (ITU-T Y.1731 Ethernet Service OAM) is not supported.
- CCM interval of less than 1 second is not supported.
- CFM is not supported on Routed Interfaces and aggregated Ethernet (lag) interfaces.
- MIP half function, to divide the MIP functionality into two unidirectional segments to improve network coverage, is not supported.
- Up MEP is not supported.

- Total number of CFM sessions supported is 20.

RELATED DOCUMENTATION

Junos OS Network Interfaces Configuration Guide

Configure Ethernet OAM Connectivity Fault Management (CLI Procedure)

IN THIS SECTION

- Creating the Maintenance Domain | 183
- Configuring the Maintenance Domain MIP Half Function | 184
- Creating a Maintenance Association | 184
- Configuring the Continuity Check Protocol | 184
- Configuring a Maintenance Association End Point | 185
- Configuring a Connectivity Fault Management Action Profile | 187
- Configuring the Linktrace Protocol | 187

Ethernet interfaces on Juniper Networks EX Series Ethernet Switches and Juniper Networks Junos OS for EX Series switches support the IEEE 802.1ag standard for Operation, Administration, and Management (OAM). The IEEE 802.1ag specification provides for Ethernet connectivity fault management (CFM).

NOTE: This feature is not supported on EX4300 switches on aggregated Ethernet (LAG) interfaces.

This topic describes these tasks:

Creating the Maintenance Domain

A maintenance domain comprises network entities such as operators, providers, and customers. To enable connectivity fault management (CFM) on an Ethernet interface, you must create a maintenance domains, maintenance associations, and MEPs.

To create a maintenance domain:

1. Specify a name for the maintenance domain:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch# set maintenance-domain domain-name
```

2. Specify a format for the maintenance domain name. If you specify `none`, no name is configured:
   - A plain ASCII character string
   - A domain name service (DNS) format
   - A media access control (MAC) address plus a two-octet identifier in the range 0 through 65,535
   - `none`

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]
user@switch# set name-format format
```

   For example, to specify the name format as MAC address plus a two-octet identifier:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]
user@switch# set name-format mac+2oct
```

3. Configure the maintenance domain level, which is used to indicate the nesting relationship between this domain and other domains. Use a value from 0 through 7:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]
user@switch# set level level
```

NOTE: The configuration display entries in the CFM maintenance domain list are "ordered by system" rather than "ordered by user."

Configuring the Maintenance Domain MIP Half Function

NOTE: MIP Half Function (MHF) is not supported on EX4600, QFX5200, and QFX5210 switches.

MIP Half Function (MHF) divides the maintenance association intermediate point (MIP) functionality into two unidirectional segments, improves visibility with minimal configuration, and improves network coverage by increasing the number of points that can be monitored. MHF extends monitoring capability by responding to loop-back and link-trace messages to help isolate faults. Whenever a MIP is configured, the MIP half function value for all maintenance domains and maintenance associations must be the same.

To configure the MIP half function:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]
user@switch# set mip-half-function  (none | default | explicit)
```

Creating a Maintenance Association

In a CFM maintenance domain, each service instance is called a maintenance association.

To create a maintenance association:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name]
user@switch# set maintenance-association ma-name
```

NOTE: The configuration display entries in the CFM maintenance domain list are "ordered by system" rather than "ordered by user."

Configuring the Continuity Check Protocol

The continuity check protocol is used for fault detection by a maintenance association end point (MEP) within a maintenance association. The MEP periodically sends continuity check multicast messages. The receiving MEPs use the continuity check messages (CCMs) to build a MEP database of all MEPs in the maintenance association.

To configure the continuity check protocol:

1. Enable the continuity check protocol:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name]
user@switch# set continuity-check
```

2. Specify the continuity check hold interval. The hold interval is the number of minutes to wait before flushing the MEP database if no updates occur. The default value is 10 minutes.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name continuity-check]
user@switch# set hold-interval number
```

3. Specify the CCM interval. The interval is the time between the transmission of CCMs. You can specify 10 minutes (10m), 1 minute (1m), 10 seconds (10s), 1 second (1s), 100 milliseconds (100ms), or 10 milliseconds (10ms).

> **NOTE**: On EX4600, QFX5200, and QFX5210 switches, CCM interval of less than 1 second is not supported.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name continuity-check]
user@switch# set interval number
```

4. Specify the number of CCMs (that is, protocol data units) that can be lost before the MEP is marked as down. The default number of protocol data units (PDUs) is 3.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name continuity-check]
user@switch# set loss-threshold number
```

Configuring a Maintenance Association End Point

To configure a maintenance association end point:

1. Specify an ID for the MEP. The value can be from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name]
user@switch# set mep mep-id]
```

2. Enable maintenance endpoint automatic discovery if you want to have the MEP accept continuity check messages (CCMs) from all remote MEPs of the same maintenance association:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id
user@switch# set auto-discovery
```

3. You can specify that CFM packets (CCMs) be transmitted only in one direction for the MEP, that is, the direction be set as `down` so that CCMs are transmitted only out of (not into) the interface configured on this MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@switch# set direction down
```

4. Specify the logical interface to which the MEP is attached. It can be either an access interface or a trunk interface. If you specify a trunk interface, the VLAN associated with that interface must have a VLAN ID.

> **NOTE**: You cannot associate an access interface that belongs to multiple VLANs with the MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@switch# set interface interface-name
```

5. You can configure a remote MEP from which CCMs are expected. If autodiscovery is not enabled, the remote MEP must be configured under the `mep` statement. If the remote MEP is not configured under the `mep` statement, the CCMs from the remote MEP are treated as errors.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain domain-name
maintenance-association ma-name mep mep-id]
user@switch# set remote-mep mep-id
```

Configuring a Connectivity Fault Management Action Profile

You can configure an action profile and specify the action to be taken when any of the configured events occur. Alternatively, you can configure an action profile and specify default actions when connectivity to a remote MEP fails.

To configure an action profile:

1. Specify a name for an action profile:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch# set action-profile profile-name
```

2. Configure the action of the action profile:

```
[edit protocols oam ethernet connectivity-fault-management action-profile profile-name]
user@switch# set action interface-down
```

3. Configure one or more events under the action profile, the occurrence of which will trigger the corresponding action to be taken:

```
[edit protocols oam ethernet connectivity-fault-management action-profile profile-name]
user@switch# set event event
```

See *Junos OS Network Interfaces Configuration Guide*

Configuring the Linktrace Protocol

The linktrace protocol is used for path discovery between a pair of maintenance points. Linktrace messages are triggered by an administrator using the `traceroute` command to verify the path between a pair of MEPs under the same maintenance association. Linktrace messages can also be used to verify the path between a MEP and a MIP under the same maintenance domain.

To configure the linktrace protocol:

1. Configure the linktrace path age timer. If no response to a linktrace request is received, the request and response entries are deleted after the age timer expires:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch# set linktrace age time
```

2. Configure the number of linktrace reply entries to be stored per linktrace request:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch# set linktrace path-database-size path-database-size
```

RELATED DOCUMENTATION

Junos OS Network Interfaces Configuration Guide

Example: Configure Ethernet OAM Connectivity Fault Management on EX Series Switches

IN THIS SECTION

- Requirements | 189
- Overview and Topology | 189
- Configuring Ethernet OAM Connectivity Fault Management on Switch 1 | 189
- Configuring Ethernet OAM Connectivity Fault Management on Switch 2 | 191
- Verification | 194

Ethernet interfaces on EX Series switches and Junos OS for EX Series switches support the IEEE 802.1ag standard for Operation, Administration, and Management (OAM). The IEEE 802.1ag specification provides for Ethernet connectivity fault management (CFM).

This example describes how to enable and configure OAM CFM on a Gigabit Ethernet interface:

Requirements

This example uses the following hardware and software components:

- Junos OS Release 10.2 or later for EX Series switches
- Two EX Series switches connected by a point-to-point Gigabit Ethernet link

Overview and Topology

CFM can be used to monitor the physical link between two switches. In the following example, two switches are connected by a point-to-point Gigabit Ethernet link. The link between these two switches is monitored using CFM.

Configuring Ethernet OAM Connectivity Fault Management on Switch 1

IN THIS SECTION

- CLI Quick Configuration | 189
- Procedure | 189
- Results | 191

CLI Quick Configuration

To quickly configure Ethernet OAM CFM, copy the following commands and paste them into the switch terminal window:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain]
set name-format character-string
set maintenance-domain private level 0
set maintenance-association private-ma
set continuity-check hold-interval 1s
```

Procedure

Step-by-Step Procedure

To enable and configure OAM CFM on switch 1:

1. Specify the maintenance domain name format:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain]
user@switch1# set name-format character-string
```

2. Specify the maintenance domain name and the maintenance domain level:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch1# set maintenance-domain private level 0
```

3. Create a maintenance association:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain private]
user@switch1# set maintenance-association private-ma
```

4. Enable the continuity check protocol and specify the continuity check hold interval:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain private
maintenance-association private-ma]
user@switch1# set continuity-check hold-interval 1s
```

5. Configure the maintenance association end point (MEP):

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain private
maintenance-association private-ma]
user@switch1# set mep 100 interface ge-1/0/1 auto-discovery direction down
```

Results

Check the results of the configuration.

```
[edit]
user@switch1 > show
```

```
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain private {
                    level 0;
                    maintenance-association private-ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 100 {
                            interface ge-1/0/1;
                            auto-discovery;
                            direction down;
                        }
                }
            }
        }
    }
```

Configuring Ethernet OAM Connectivity Fault Management on Switch 2

IN THIS SECTION

- CLI Quick Configuration | **192**
- Procedure | **192**
- Results | **193**

CLI Quick Configuration

To quickly configure Ethernet OAM CFM, copy the following commands and paste them into the switch terminal window:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain]
set name-format character-string
set maintenance-domain private level 0
set maintenance-association private-ma
set continuity-check hold-interval 1s
```

Procedure

Step-by-Step Procedure

The configuration on switch 2 mirrors that on switch 2.

1. Specify the maintenance domain name format:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch2# set name-format character-string
```

2. Specify the maintenance domain name and the maintenance domain level:

```
[edit protocols oam ethernet connectivity-fault-management]
user@switch2# set maintenance-domain private level 0
```

3. Create a maintenance association:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain private]
user@switch2# set maintenance-association private-ma
```

4. Enable the continuity check protocol and specify the continuity check hold interval:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain private
maintenance-association private-ma]
user@switch2# set continuity-check hold-interval 1s
```

5. Configure the maintenance association end point (MEP)

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain private
maintenance-association private-ma]
user@switch2# set mep 200 interface ge-0/2/5 auto-discovery direction down
```

Results

Check the results of the configuration.

```
[edit]
user@switch2 > show
```

```
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain private {
                    level 0;
                    maintenance-association private-ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 200 {
                            interface ge-0/2/5;
                            auto-discovery;
                            direction down;
                        }
                }
            }
        }
    }
```

Verification

IN THIS SECTION

- Verifying That OAM CFM Has Been Configured Properly | **194**

To confirm that the configuration is working properly, perform these tasks:

Verifying That OAM CFM Has Been Configured Properly

Purpose

Verify that OAM CFM has been configured properly.

Action

Use the `show oam ethernet connectivity-fault-management interfaces detail` command:

```
user@switch1# show oam ethernet connectivity-fault-management interfaces detail
```

Sample Output

command-name

```
Interface name: ge-1/0/1.0, Interface status: Active, Link status: Up
  Maintenance domain name: private, Format: string, Level: 0
    Maintenance association name: private-ma, Format: string
    Continuity-check status: enabled, Interval: 1ms, Loss-threshold: 3 frames
    MEP identifier: 100, Direction: down, MAC address: 00:90:69:0b:4b:94
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                 : no
      Erroneous CCM received                       : yes
      Cross-connect CCM received                   : no
      RDI sent by some MEP                         : yes
    Statistics:
      CCMs sent                                    : 76
```

```
    CCMs received out of sequence                : 0
    LBMs sent                                    : 0
    Valid in-order LBRs received                 : 0
    Valid out-of-order LBRs received             : 0
    LBRs received with corrupted data            : 0
    LBRs sent                                    : 0
    LTMs sent                                    : 0
    LTMs received                                : 0
    LTRs sent                                    : 0
    LTRs received                                : 0
    Sequence number of next LTM request          : 0
  Remote MEP count: 2
    Identifier    MAC address        State    Interface
     2001     00:90:69:0b:7f:71       ok    ge-0/2/5.0
```

Meaning

When the output displays that continuity-check status is `enabled` and displays details of the remote MEP, it means that connectivity fault management (CFM) has been configured properly.

RELATED DOCUMENTATION

- Junos OS Network Interfaces Configuration Guide

CHAPTER 5

Ethernet Frame Delay

IN THIS CHAPTER

- Ethernet Frame Delay Measurements on Switches | **196**
- Configure MEP Interfaces on Switches to Support Ethernet Frame Delay Measurements (CLI Procedure) | **198**
- Configure One-Way Ethernet Frame Delay Measurements on Switches (CLI Procedure) | **199**
- Configure an Iterator Profile on a Switch (CLI Procedure) | **200**
- Trigger an Ethernet Frame Delay Measurement Session on a Switch | **201**
- Configure Two-Way Ethernet Frame Delay Measurements on Switches (CLI Procedure) | **202**

Ethernet Frame Delay Measurements on Switches

IN THIS SECTION

- Ethernet Frame Delay Measurements | **197**
- Types of Ethernet Frame Delay Measurements | **197**
- Limitations | **198**

In many cases, a service provider could be subject to penalties imposed by regulation, statute, or contract if network performance is not within the bounds established for the service. One key performance objective is delay, along with its close relative, delay variation (often called *jitter*). Some applications (such as bulk file transfer) will function just as well with high delays across the network and high delay variations, while other applications (such as voice) can function only with low and stable delays. Many networks invoke protocols or features available at Layer 3 (the packet layer) or higher to measure network delays and jitter link by link. However, when the network consists of many Ethernet links, there are few protocols and features available at Layer 2 (the frame layer) that allow routers and

switches to measure frame delay and jitter. This is where the ability to configure and monitor Ethernet frame delay is helpful.

This topic includes:

Ethernet Frame Delay Measurements

You can perform Ethernet frame delay measurements (referred to as ETH-DM in Ethernet specifications) on Juniper Networks EX Series Ethernet Switches. This feature allows you to configure on-demand Operation, Administration, and Maintenance (OAM) statements for the measurement of frame delay and frame delay variation (jitter). You can configure Ethernet frame delay measurement in either one-way or two-way (round-trip) mode to gather frame delay statistics simultaneously from multiple sessions. Ethernet frame delay measurement provides fine control to operators for triggering delay measurement on a given service and can be used to monitor SLAs.

Ethernet frame delay measurement also collects other useful information, such as worst and best case delays, average delay, and average delay variation. It supports software-assisted timestamping in the receive direction for delay measurements. It also provides runtime display of delay statistics when two-way delay measurement is triggered. Ethernet frame delay measurement records the last 100 samples collected per remote maintenance association end point (MEP) or per connectivity fault management (CFM) session. You can retrieve the history at any time using simple commands. You can clear all Ethernet frame delay measurement statistics and PDU counters. Ethernet frame delay measurement is fully compliant with the ITU-T Y.1731 (*OAM Functions and Mechanisms for Ethernet-based Networks*) specification.

Ethernet frame delay measurement uses the IEEE 802.1ag CFM infrastructure.

Generally, Ethernet frame delay measurements are made in a peer fashion from one MEP or CFM session to another. However, these measurements are not made to maintenance association intermediate points (MIPs).

For a complete description of Ethernet frame delay measurement, see the *ITU-T Y.1731 Ethernet Service OAM* topics in the Junos OS Network Interfaces Library for Routing Devices.

Types of Ethernet Frame Delay Measurements

There are two types of Ethernet frame delay measurements:

- One-way
- Two-way (round-trip)

For one-way Ethernet frame delay measurement, either MEP can send a request to begin a one-way delay measurement to its peer MEP. However, the statistics are collected only at the receiver MEP. This feature requires the clocks at the transmitting and receiving MEPs to be synchronized. If these clocks fall

out of synchronization, only one-way delay variation and average delay variation values are computed correctly (and will, therefore, be valid). Use the `show` commands at the receiver MEP to display one-way delay statistics.

For two-way (round-trip) Ethernet frame delay measurement, either MEP can send a request to begin a two-way delay measurement to its peer MEP, which responds with timestamp information. Run-time statistics are collected and displayed at the initiator MEP. The clocks do not need to be synchronized at the transmitting and receiving MEPs. Junos OS supports timestamps in delay measurement reply (DMR) frames to increase the accuracy of delay calculations.

Use the `show` commands at the initiator MEP to display two-way delay statistics, and at the receiver MEP to display one-way delay statistics.

You can create an iterator profile to periodically transmit SLA measurement packets in the form of ITU-Y.1731-compliant frames for delay measurement or loss measurement.

Limitations

The following are some limitations with regard to using Ethernet frame delay measurement:

- Ethernet frame delay measurements are available only when distributed periodic packet management (PPM) is enabled.
- The statistics collected are lost after a *graceful Routing Engine switchover* (GRES).
- You can monitor only one session to the same remote MEP or MAC address.
- Accuracy is compromised when the system configuration changes (such as from reconfiguration). We recommend performing Ethernet frame delay measurements on a stable system.

Configure MEP Interfaces on Switches to Support Ethernet Frame Delay Measurements (CLI Procedure)

Ethernet frame delay measurement is a useful tool for providing performance statistics or supporting or challenging service-level agreements (SLAs). By default, Ethernet frame delay measurement uses software for timestamping and delay calculations. You can configure an EX Series switch to perform and display Ethernet frame delay measurements on Ethernet interfaces. The switches support software-assisted timestamping.

Before you can begin configuring MEP interfaces to support Ethernet frame delay measurements on switches, ensure that you have:

- Configured Operation, Administration, and Maintenance (OAM) connectivity fault management (CFM) correctly
- Enabled distributed periodic packet management (PPM) (distributed PPM is enabled by default)

To configure MEP interfaces on switches to support Ethernet frame delay measurements:

Enable the Ethernet frame delay measurement by issuing the **monitor ethernet delay-measurement** operational mode command. In this command, you must specify one measurement type (either one-way or two-way measurement), and you must specify either the unicast MAC address of the peer MEP or its numeric identifier.

Optionally, you can also specify the following parameters:

- Number of frames to send to the peer MEP (**count** ***count***)
- Number of seconds to wait between sending frames (**wait** ***time***)
- Priority value of the delay measurement request frame (**priority** ***value***)
- Size of the data in the data TLV of the request packet (**size** ***value***)
- Suppression of the insertion of the session ID TLV in the request packet (**no-session-id-tlv**)

```
user@switch> monitor ethernet delay-measurement  maintenance-domain md-name  maintenance-
association ma-name  one-way mep remote-mep-id count count wait time priority value size value
no-session-id-tlv
```

Configure One-Way Ethernet Frame Delay Measurements on Switches (CLI Procedure)

Ethernet frame delay measurement is a useful tool for providing performance statistics or supporting or challenging service-level agreements (SLAs). You can configure the frame delay measurements in either a one-way mode or a two-way (round-trip) mode to gather frame delay statistics. For one-way Ethernet frame delay measurement, clocks at the local and remote MEPs need to be synchronized. However, clock synchronization is not required for two-way Ethernet frame delay measurement.

Before you begin configuring one-way Ethernet frame delay measurements on two EX Series switches, ensure that you have:

- Configured Operation, Administration, and Maintenance (OAM) connectivity fault management (CFM) correctly on both the switches

- Synchronized the system clocks of both the switches

To configure one-way Ethernet frame delay measurements:

1. Configure the maintenance domain, maintenance association, and MEP ID on both the switches.
2. From either switch, start a one-way Ethernet frame delay measurement:

```
user@switch> monitor ethernet delay-measurement  maintenance-domain md-name  maintenance-
association ma-name  one-way mep remote-mep-id count count wait time
```

You can view the result on the other switch:

```
user@switch> show oam ethernet connectivity-fault-management delay-statistics maintenance-domain
md-name  maintenance-association ma-name local-mep mep-id remote-mep mep-id
```

Configure an Iterator Profile on a Switch (CLI Procedure)

Ethernet frame delay measurement provides fine control to operators for triggering delay measurement on a given service and can be used to monitor service-level agreements (SLAs). You can create an iterator profile with its parameters to periodically transmit SLA measurement packets in the form of ITU-Y.1731-compliant frames for two-way delay measurement.

To create an iterator profile:

1. Specify a name for an SLA iterator profile—for example, i1:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@switch# edit sla-iterator-profiles i1
```

2. (Optional) Configure the cycle time, which is the time (in milliseconds) between back-to-back transmissions of SLA frames.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@switch# set cycle-time cycle-time-value
```

3. (Optional) Configure the iteration period, which indicates the maximum number of cycles per iteration (the number of connections registered to an iterator cannot exceed this value).

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@switch# set iteration-period iteration-period-value
```

4. Configure the measurement type as two-way delay measurement.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@switch# set measurement-type two-way-delay
```

5. (Optional) Configure the calculation weight for delay.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@switch# set calculation-weight delay delay-value
```

6. (Optional) Configure the calculation weight for delay variation.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@switch# set calculation-weight delay-variation delay-variation-value
```

7. Configure a remote MEP with the iterator profile.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id]
user@switch# set sla-iterator-profiles i1
```

Trigger an Ethernet Frame Delay Measurement Session on a Switch

To trigger Ethernet frame delay measurement, use the *monitor ethernet delay-measurement* operational command and specify the following values:

- Either one-way (`one-way`) or two-way (`two-way`) measurement

- Either the MAC address (`remote-mac-address`) or the MEP ID (`mep`) of the remote host
- The maintenance domain (`maintenance-domain`)
- The maintenance association (`maintenance-association`)
- (Optional) Any or all of these options: `count`, `size`, `wait`, no-session-id-tlv, `priority`

For example:

```
user@switch> monitor ethernet delay-measurement one-way 00:05:85:73:39:4a maintenance-domain md6
maintenance-association ma6 count 10 size 50 wait 5 no-session-id-tlv priority 1
```

Configure Two-Way Ethernet Frame Delay Measurements on Switches (CLI Procedure)

Ethernet frame delay measurement is a useful tool for providing performance statistics or supporting or challenging service-level agreements (SLAs). You can configure the frame delay measurements in either a one-way mode or a two-way (round-trip) mode to gather frame delay statistics. For one-way Ethernet frame delay measurement, clocks at the local and remote MEPs need to be synchronized. However, clock synchronization is not required for two-way Ethernet frame delay measurement.

Before you begin configuring two-way Ethernet frame delay measurements on two EX Series switches, ensure that you have:

- Configured Operation, Administration, and Maintenance (OAM) connectivity fault management (CFM) correctly on both the switches

To configure two-way Ethernet frame delay measurements:

1. Configure the maintenance domain, maintenance association, and MEP ID on both the switches.
2. From either switch, start a two-way Ethernet frame delay measurement:

```
user@switch> monitor ethernet delay-measurement  maintenance-domain md-name  maintenance-
association ma-name  two-way mep remote-mep-id count count wait time
```

You can view the result on the other switch:

```
user@switch> show oam ethernet connectivity-fault-management delay-statistics maintenance-domain
md-name  maintenance-association ma-name local-mep mep-id remote-mep mep-id
```

CHAPTER 6

Ethernet Service OAM (ITU-TY.1731) for Routers

IN THIS CHAPTER

- ITU-T Y.1731 Ethernet Service OAM Overview | 204
- Configure Ethernet Frame Delay Measurement Sessions | 224
- Configuring MEP Interfaces to Support Ethernet Frame Delay Measurements | 266
- Configure Ethernet Frame Loss Measurement | 268
- Configure an Iterator Profile | 305
- Configure Ethernet Synthetic Loss Measurements | 324
- Ethernet Alarm Indication | 341
- Inline Transmission Mode | 356

ITU-T Y.1731 Ethernet Service OAM Overview

SUMMARY

This section describes service OAM (ITU-TY.1731) and its two main components: fault management (monitoring, detection, and isolation) and performance monitoring (frame loss measurement, synthetic frame loss measurement, and frame delay measurement).

IN THIS SECTION

- Ethernet Frame Delay Measurements Overview | 205
- Ethernet Frame Loss Measurement Overview | 212
- Service-Level Agreement Measurement | 213
- On-Demand Mode for SLA Measurement | 213
- Proactive Mode for SLA Measurement | 214
- Ethernet Failure Notification Protocol Overview | 216

- Ethernet Synthetic Loss Measurement Overview | **217**
- Scenarios for Configuration of ETH-SLM | **218**
- Format of ETH-SLM Messages | **219**
- Transmission of ETH-SLM Messages | **221**

Ethernet Frame Delay Measurements Overview

IN THIS SECTION

- ITU-T Y.1731 Frame Delay Measurement Feature | **205**
- One-Way Ethernet Frame Delay Measurement | **208**
- Two-Way Ethernet Frame Delay Measurement | **209**
- Choosing Between One-Way and Two-Way ETH-DM | **210**
- Restrictions for Ethernet Frame Delay Measurement | **211**

ITU-T Y.1731 Frame Delay Measurement Feature

The IEEE 802.3-2005 standard for Ethernet Operations, Administration, and Maintenance (OAM) defines a set of link fault management mechanisms to detect and report link faults on a single point-to-point Ethernet LAN.

Junos OS supports key OAM standards that provide for automated end-to-end management and monitoring of Ethernet service by service providers:

- *IEEE Standard 802.1ag*, also known as “Connectivity Fault Management (CFM).”
- *ITU-T Recommendation Y.1731*, which uses different terminology than IEEE 802.1ag and defines Ethernet service OAM features for fault monitoring, diagnostics, and performance monitoring.

These capabilities allow operators to offer binding service-level agreements (SLAs) and generate new revenues from rate- and performance-guaranteed service packages that are tailored to the specific needs of their customers.

ACX Series routers support proactive and on-demand modes.

You can configure ITU-T Y.1731 standard-compliant Ethernet loss measurement (ETH-LM), Ethernet synthetic loss measurement (ETH-SLM), and Ethernet delay measurement (ETH- DM) capabilities on MPC10 and MPC11 line cards on 20.2R2-S3 only and 20.4R1 onwards.

NOTE: ACX5048 and ACX5096 routers supports only software-based time stamping for delay measurement.

Ethernet CFM

The IEEE 802.1ag standard for connectivity fault management (CFM) defines mechanisms to provide for end-to-end Ethernet service assurance over any path, whether a single link or multiple links spanning networks composed of multiple LANs.

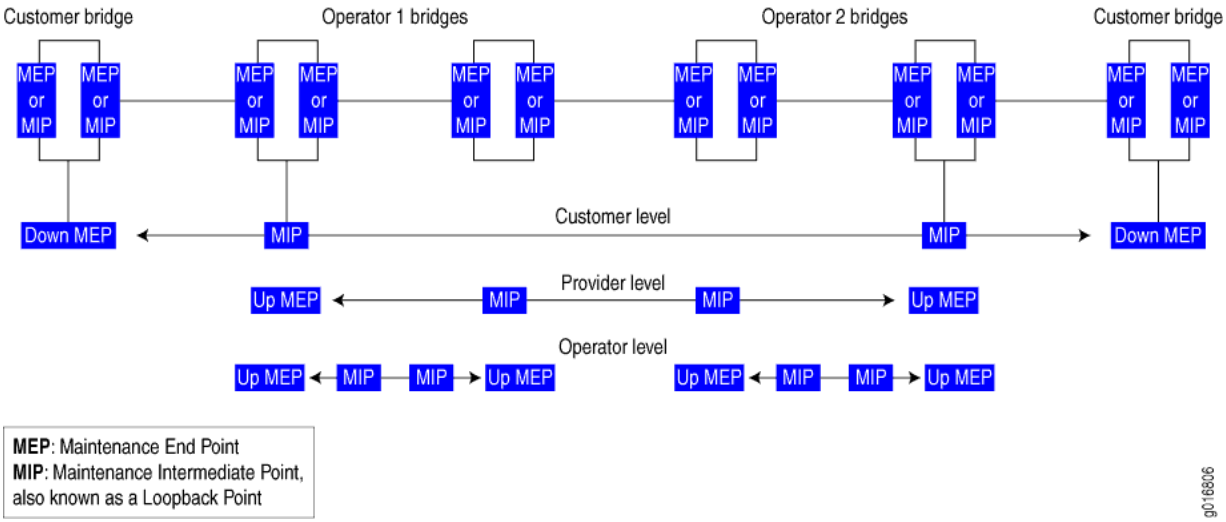
For Ethernet interfaces on M320, MX Series, and T Series routers, Junos OS supports the following key elements of the Ethernet CFM standard:

- Fault monitoring using the IEEE 802.1ag Ethernet OAM Continuity Check protocol
- Path discovery and fault verification using the IEEE 802.1ag Ethernet OAM Linktrace protocol
- Fault isolation using the IEEE 802.1ag Ethernet OAM Loopback protocol

In a CFM environment, network entities such as network operators, service providers, and customers may be part of different administrative domains. Each administrative domain is mapped into one maintenance domain. Maintenance domains are configured with different level values to keep them separate. Each domain provides enough information for the entities to perform their own management and end-to-end monitoring, and still avoid security breaches.

Figure 18 on page 207 shows the relationships among the customer, provider, and operator Ethernet bridges, maintenance domains, maintenance association end points (MEPs), and maintenance intermediate points (MIPs).

Figure 18: Relationship of MEPs, MIPs, and Maintenance Domain Levels



NOTE: On ACX Series routers, the maintenance intermediate points (MIP) is supported only on the ACX5048 and ACX5096 routers.

Ethernet Frame Delay Measurement

Two key objectives of OAM functionality are to measure quality-of-service attributes such as frame delay and frame delay variation (also known as “frame *jitter*”). Such measurements can enable you to identify network problems before customers are impacted by network defects.

Junos OS supports Ethernet frame delay measurement between MEPs configured on Ethernet physical or logical interfaces on MX Series routers. Ethernet frame delay measurement provides fine control to operators for triggering delay measurement on a given service and can be used to monitor SLAs. Ethernet frame delay measurement also collects other useful information, such as worst and best case delays, average delay, and average delay variation. The Junos OS implementation of Ethernet frame delay measurement (ETH-DM) is fully compliant with the ITU-T Recommendation Y.1731, *OAM Functions and Mechanisms for Ethernet-based Networks*. The recommendation defines OAM mechanisms for operating and maintaining the network at the Ethernet service layer, which is called the "ETH layer" in ITU-T terminology.

MX Series routers with modular port concentrators (MPCs) and 10-Gigabit Ethernet MPCs with SFP+ support ITU-T Y.1731 functionality on VPLS for frame-delay and delay-variation.

NOTE: MX Series Virtual Chassis does not support Ethernet frame delay measurement (DM).

One-Way Ethernet Frame Delay Measurement

In one-way ETH-DM mode, a series of frame delay and frame delay variation values are calculated based on the time elapsed between the time a measurement frame is sent from the initiator MEP at one router and the time when the frame is received at the receiver MEP at the other router.

NOTE: ACX Series routers do not support one-way Ethernet frame delay measurement.

1DM Transmission

When you start a one-way frame delay measurement, the router sends 1DM frames—frames that carry the protocol data unit (PDU) for a one-way delay measurement—from the initiator MEP to the receiver MEP at the rate and for the number of frames you specify. The router marks each 1DM frame as drop-ineligible and inserts a timestamp of the transmission time into the frame.

1DM Reception

When an MEP receives a 1DM frame, the router that contains the receiver MEP measures the one-way delay for that frame (the difference between the time the frame was received and the timestamp contained in the frame itself) and the delay variation (the difference between the current and previous delay values).

One-Way ETH-DM Statistics

The router that contains the receiver MEP stores each set of one-way delay statistics in the ETH-DM database. The ETH-DM database collects up to 100 sets of statistics for any given CFM session (pair of peer MEPs). You can access these statistics at any time by displaying the ETH-DM database contents.

One-Way ETH-DM Frame Counts

Each router counts the number of one-way ETH-DM frames sent and received:

- For an initiator MEP, the router counts the number of 1DM frames sent.
- For a receiver MEP, the router counts the number of valid 1DM frames received and the number of invalid 1DM frames received.

Each router stores ETH-DM frame counts in the CFM database. The CFM database stores CFM session statistics and, for interfaces that support ETH-DM, any ETH-DM frame counts. You can access the frame counts at any time by displaying CFM database information for Ethernet interfaces assigned to MEPs or for MEPs in CFM sessions.

Synchronization of System Clocks

The accuracy of one-way delay calculations depends on close synchronization of the system clocks at the initiator MEP and receiver MEP.

The accuracy of one-way delay variation is not dependent on system clock synchronization. Because delay variation is simply the difference between consecutive one-way delay values, the out-of-phase period is eliminated from the frame jitter values.

NOTE: For a given one-way Ethernet frame delay measurement, frame delay and frame delay variation values are available only on the router that contains the receiver MEP.

Two-Way Ethernet Frame Delay Measurement

In two-way ETH-DM mode, frame delay and frame delay variation values are based on the time difference between when the initiator MEP transmits a request frame and receives a reply frame from the responder MEP, subtracting the time elapsed at the responder MEP.

DMM Transmission

When you start a two-way frame delay measurement, the router sends delay measurement message (DMM) frames— frames that carry the PDU for a two-way ETH-DM request—from the initiator MEP to the responder MEP at the rate and for the number of frames you specify. The router marks each DMM frame as drop-ineligible and inserts a timestamp of the transmission time into the frame.

DMR Transmission

When an MEP receives a DMM frame, the responder MEP responds with a delay measurement reply (DMR) frame, which carries ETH-DM reply information and a copy of the timestamp contained in the DMM frame.

DMR Reception

When an MEP receives a valid DMR, the router that contains the MEP measures the two-way delay for that frame based on the following sequence of timestamps:

1. TI_{TxDMM}
2. TR_{RxDMM}
3. TR_{TxDMR}

4. TI_{RxDMR}

A two-way frame delay is calculated as follows:

1. $[TI_{RxDMR} - TI_{TxDMM}] - [TR_{TxDMR} - TR_{RxDMM}]$

The calculation show that frame delay is the difference between the time at which the initiator MEP sends a DMM frame and the time at which the initiator MEP receives the associated DMR frame from the responder MEP, minus the time elapsed at the responder MEP.

The delay variation is the difference between the current and previous delay values.

Two-Way ETH-DM Statistics

The router that contains the initiator MEP stores each set of two-way delay statistics in the ETH-DM database. The ETH-DM database collects up to 100 sets of statistics for any given CFM session (pair of peer MEPs). You can access these statistics at any time by displaying the ETH-DM database contents.

Two-Way ETH-DM Frame Counts

Each router counts the number of two-way ETH-DM frames sent and received:

- For an initiator MEP, the router counts the number DMM frames transmitted, the number of valid DMR frames received, and the number of invalid DMR frames received.
- For a responder MEP, the router counts the number of DMR frames sent.

Each router stores ETH-DM frame counts in the CFM database. The CFM database stores CFM session statistics and, for interfaces that support ETH-DM, any ETH-DM frame counts. You can access the frame counts at any time by displaying CFM database information for Ethernet interfaces assigned to MEPs or for MEPs in CFM sessions.

> **NOTE**: For a given two-way Ethernet frame delay measurement, frame delay and frame delay variation values are available only at the router that contains the initiator MEP.

Choosing Between One-Way and Two-Way ETH-DM

One-way frame delay measurement requires that the system clocks at the initiator MEP and receiver MEP are closely synchronized. Two-way frame delay measurement does not require synchronization of the two systems. If it is not practical for the clocks to be synchronized, two-way frame delay measurements are more accurate.

When two systems are physically close to each other, their one-way delay values are very high compared to their two-way delay values. One-way delay measurement requires that the timing for the two systems be synchronized at a very granular level, and MX Series routers currently do not support this granular synchronization.

Restrictions for Ethernet Frame Delay Measurement

The following restrictions apply to the Ethernet frame delay measurement feature:

- The ETH-DM feature is not supported on label-switched interface (LSI) pseudowires.

  The ETH-DM feature is supported on aggregated Ethernet interfaces.

- Hardware-assisted timestamping for ETH-DM frames in the reception path is only supported for MEP interfaces on Enhanced DPCs and Enhanced Queuing DPCs in MX Series routers. For information about hardware-assisted timestamping, see "Guidelines for Configuring Routers to Support an ETH-DM Session" on page 224 and "Enabling the Hardware-Assisted Timestamping Option" on page 224.

- Ethernet frame delay measurements can be triggered only when the distributed periodic packet management daemon (`ppm`) is enabled. For more information about this limitation, see "Guidelines for Configuring Routers to Support an ETH-DM Session" on page 224 and "Ensuring That Distributed ppm Is Not Disabled" on page 234.

- You can monitor only one session at a time to the same remote MEP or MAC address. For more information about starting an ETH-DM session, see "Starting an ETH-DM Session" on page 241.

- ETH-DM statistics are collected at only one of the two peer routers in the ETH-DM session. For a one-way ETH-DM session, you can display frame ETH-DM statistics at the receiver MEP only, using ETH-DM-specific `show` commands. For a two-way ETH-DM session, you can display frame delay statistics at the initiator MEP only, using the same ETH-DM-specific `show` commands. For more information, see "Managing ETH-DM Statistics and ETH-DM Frame Counts" on page 259.

- ETH-DM frame counts are collected at both MEPs and are stored in the respective CFM databases.

- If *graceful Routing Engine switchover* (GRES) occurs, any collected ETH-DM statistics are lost, and ETH-DM frame counts are reset to zeroes. Therefore, the collection of ETH-DM statistics and ETH-DM frame counters has to be restarted, after the switchover is complete. GRES enables a router with dual Routing Engines to switch from a primary Routing Engine to a backup Routing Engine without interruption to packet forwarding. For more information, see the Junos OS High Availability User Guide.

- Accuracy of frame delay statistics is compromised when the system is changing (such as from reconfiguration). We recommend performing Ethernet frame delay measurements on a stable system.

Ethernet Frame Loss Measurement Overview

The key objectives of the OAM functionality are to measure quality-of-service attributes such as frame delay, frame delay variation (also known as “frame *jitter*”), and frame loss. Such measurements enable you to identify network problems before customers are impacted by network defects.

Junos OS supports Ethernet frame loss measurement (ETH-LM) between maintenance association end points (MEPs) configured on Ethernet physical or logical interfaces on MX Series routers and is presently supported only for *VPWS* service. ETH-LM is used by operators to collect counter values applicable for ingress and egress service frames. These counters maintain a count of transmitted and received data frames between a pair of MEPs. Ethernet frame loss measurement is performed by sending frames with ETH-LM information to a peer MEP and similarly receiving frames with ETH-LM information from the peer MEP. This type of frame loss measurement is also known as single-ended Ethernet loss measurement.

NOTE: MX Series Virtual Chassis does not support Ethernet frame loss measurement (ETH-LM).

ETH-LM supports the following frame loss measurements:

- Near-end frame loss measurement—Measurement of frame loss associated with ingress data frames.
- Far-end frame loss measurement—Measurement of frame loss associated with egress data frames.

NOTE: The proactive and dual-ended loss measurement functionality of ITU-T Y1731 is not supported on the ACX Series routers.

The ETH-LM feature is supported on aggregated Ethernet interfaces.

NOTE: Starting Junos OS Release 16.1, the Ethernet loss measurement (ETH-LM) results are inaccurate when connectivity fault management (CFM) and performance monitoring (PM) PDUs received locally at a maintenance endpoint (MEP) as classified as belonging to the yellow class or a packet loss priority (PLP) of medium-high. This problem of incorrect results is specific to Ethernet loss measurement for CFM sessions of down MEPs. The Ethernet loss measurement statistics are inaccurate in the following scenarios:

- Ethernet loss measurement is working on a CFM session for a MEP in down state
- CFM PDUs received on the logical interface of the down MEP are classified by the classifier as yellow or medium-high PLP

- A packet is identified as yellow when the input classifier marks the PLP as medium-high.

The problem of discrepancies with Ethernet loss measurement results is not observed when you configure Ethernet loss measurement in colorless mode. To avoid this problem of inaccurate loss measurement results, provision all local CFM PDUs as green or with the PLP as high.

NOTE: Starting with Junos OS Release 16.1, performance monitoring for connectivity fault management (by including the `performance-monitoring` statement and its substatements at the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level) is not supported when the network-to-network (NNI) or egress interface is an aggregated Ethernet interface with member links on DPCs.

Service-Level Agreement Measurement

Service-level agreement (SLA) measurement is the process of monitoring the bandwidth, delay, delay variation (*jitter*), continuity, and availability of a service (E-Line or E-LAN). It enables you to identify network problems before customers are impacted by network defects.

NOTE: The Ethernet VPN services can be classified into:

- Peer-to-peer-services (E-Line services)—The E-Line services are offered using MPLS-based Layer 2 VPN *virtual private wire service* (*VPWS*).

- Multipoint-to-multipoint services (E-LAN services)—The E-LAN services are offered using MPLS-based virtual private LAN service (VPLS).

For more information, see the *Junos VPNs Configuration Guide*.

In Junos OS, SLA measurements are classified into:

- On-demand mode—In on-demand mode, the measurements are triggered through the CLI.
- Proactive mode—In proactive mode, the measurements are triggered by an iterator application.

Note that Ethernet frame delay measurement and Ethernet frame loss measurement are not supported on the `ae` interface.

On-Demand Mode for SLA Measurement

In on-demand mode, the measurements are triggered by the user through the CLI.

When the user triggers the delay measurement through the CLI, the delay measurement request that is generated is as per the frame formats specified by the ITU-T Y.1731 standard. For two-way delay measurement, the server-side processing can be delegated to the Packet Forwarding Engine to prevent overloading on the Routing Engine. For more information, see "Configuring Routers to Support an ETH-DM Session" on page 234. When the server-side processing is delegated to the Packet Forwarding Engine, the delay measurement message (DMM) frame `receive` counters and delay measurement reply (DMR) frame `transmit` counters are not displayed by the `show` command.

When the user triggers the loss measurement through the CLI, the router sends the packets in standard format along with the loss measurement TLV. By default, the `session-id-tlv` argument is included in the packet to allow concurrent loss measurement sessions from same local MEP. You can also disable the session ID TLV by using the `no-session-id-tlv` argument.

Single-ended ETH-LM is used for on-demand operation, administration, and maintenance purposes. An MEP sends frames with ETH-LM request information to its peer MEP and receives frames with ETH-LM reply information from its peer MEP to carry out loss measurements. The protocol data unit (PDU) used for a single-ended ETH-LM request is referred to as a loss measurement message (LMM) and the PDU used for a single-ended ETH-LM reply is referred to as a loss measurement reply (LMR).

Proactive Mode for SLA Measurement

IN THIS SECTION

- Ethernet Delay Measurements and Loss Measurement by Proactive Mode | **215**

In proactive mode, SLA measurements are triggered by an iterator application. An iterator is designed to periodically transmit SLA measurement packets in form of ITU-Y.1731-compliant frames for two-way delay measurement or loss measurement on MX Series routers. This mode differs from on-demand SLA measurement, which is user initiated. The iterator sends periodic delay or loss measurement request packets for each of the connections registered to it. Iterators make sure that measurement cycles do not occur at the same time for the same connection to avoid CPU overload. Junos OS supports proactive mode for *VPWS*. For an iterator to form a remote adjacency and to become functionally operational, the continuity check message (CCM) must be active between the local and remote MEP configurations of the connectivity fault management (CFM). Any change in the iterator adjacency parameters resets the existing iterator statistics and restarts the iterator. Here, the term adjacency refers to a pairing of two endpoints (either connected directly or virtually) with relevant information for mutual understanding, which is used for subsequent processing. For example, the iterator adjacency refers to the iterator association between the two endpoints of the MEPs.

For every DPC or MPC, only 30 iterator instances for a cycle time value of 10 milliseconds (ms) are supported. In Junos OS, 255 iterator profile configurations and 2000 remote MEP associations are supported.

Iterators with cycle time value less than 100 ms are supported only for infinite iterators, whereas the iterators with cycle time value greater than 100 ms are supported for both finite and infinite iterators. Infinite iterators are iterators that run infinitely until the iterator is disabled or deactivated manually.

NOTE: ACX5048 and ACX5096 routers supports iterator cycle time of only 1 second and above.

A VPWS service configured on a router is monitored for SLA measurements by registering the connection (here, the connection is a pair of remote and local MEPs) on an iterator and then initiating periodic SLA measurement frame transmission on those connections. The end-to-end service is identified through a maintenance association end point (MEP) configured at both ends.

For two-way delay measurement and loss measurement, an iterator sends a request message for the connection in the list (if any) and then sends a request message for the connection that was polled in the former iteration cycle. The back-to-back request messages for the SLA measurement frames and their responses help in computing delay variation and loss measurement.

The Y.1731 frame transmission for a service attached to an iterator continues endlessly unless intervened and stopped by an operator or until the iteration-count condition is met. To stop the iterator from sending out any more proactive SLA measurement frames, the operator must perform one of the following tasks:

- Enable the `deactivate sla-iterator-profile` statement at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`md-name`* `maintenance association` *`ma-name`* `mep` *`mep-id`* `remote-mep` *`mep-id`*`]` hierarchy level.
- Provision a `disable` statement under the corresponding iterator profile at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-profiles` *`profile-name`*`]` hierarchy level.

Ethernet Delay Measurements and Loss Measurement by Proactive Mode

In two-way delay measurement, the delay measurement message (DMM) frame is triggered through an iterator application. The DMM frame carries an iterator type, length, and value (TLV) in addition to the fields described in standard frame format and the server copies the iterator TLV from the DMM frame to the delay measurement reply (DMR) frame.

In one-way delay variation computation using the two-way delay measurement method, the delay variation computation is based on the timestamps that are present in the DMR frame (and not the 1DM frame). Therefore, there is no need for client-side and server-side clocks to be in sync. Assuming that the

difference in their clocks remains constant, the one-way delay variation results are expected to be fairly accurate. This method also eliminates the need to send separate 1DM frames just for the one-way delay variation measurement purpose.

In proactive mode for loss measurement, the router sends packets in standard format along with loss measurement TLV and iterator TLV.

Ethernet Failure Notification Protocol Overview

The Failure Notification Protocol (FNP) is a failure notification mechanism that detects failures in Point-to-Point Ethernet transport networks on MX Series routers. If a node link fails, FNP detects the failure and sends out FNP messages to the adjacent nodes that a circuit is down. Upon receiving the FNP message, nodes can redirect traffic to the protection circuit.

NOTE: FNP is supported on E-Line services only.

An E-Line service provides a secure Point-to-Point Ethernet connectivity between two user network interfaces (UNIs). E-Line services are a protected service and each service has a working circuit and protection circuit. CFM is used to monitor the working and protect paths. CCM intervals result in failover time in hundreds of milliseconds or a few seconds. FNP provides service circuit failure detection and propagation in less than 50ms and provide 50ms failover for E-Line services.

The MX router acts as a PE node and handles the FNP messages received on the management VLAN and the FNP messages received on both the Ethernet interfaces and PWs created for the management VPLS. MX-series routers do not initiate FNP messages and responds only to FNP messages generated by devices in the Ethernet Access network. FNP can be enabled only on logical interfaces that are part of a VPLS routing instance, and no physical interfaces in that VPLS routing instance should have CCM configured. FNP can be enabled only on one *logical interface* per physical interface.

All E-Line services are configured as layer 2 circuits with edge protection. A VLAN associated with the working circuit or protection circuit must map to a logical interface. No trunk port or access port is supported in the ring link for VLANs used by E-LINE services. FNP does not control the logical interface associated with protection circuit. Only E-Line service whose termination point is not in an MX node is controlled by FNP.

FNP supports graceful restart and the *Graceful Routing Engine switchover* (GRES) features.

SEE ALSO

- *show oam ethernet fnp interface*
- *show oam ethernet fnp status*
- *show oam ethernet fnp messages*

connectivity-fault-management

Ethernet Synthetic Loss Measurement Overview

Ethernet synthetic loss measurement (ETH-SLM) is an application that enables the calculation of frame loss by using synthetic frames instead of data traffic. This mechanism can be considered as a statistical sample to approximate the frame loss ratio of data traffic. Each maintenance association end point (MEP) performs frame loss measurements, which contribute to unavailable time.

A near-end frame loss specifies frame loss associated with ingress data frames and a far-end frame loss specifies frame loss associated with egress data frames. Both near-end and far-end frame loss measurements contribute to near-end severely errored seconds and far-end severely errored seconds that are used in combination to determine unavailable time. ETH-SLM is performed using synthetic loss message (SLM) and synthetic loss reply (SLR) frames. ETH-SLM facilitates each MEP to perform near-end and far-end synthetic frame loss measurements by using synthetic frames because a bidirectional service is defined as unavailable if either of the two directions is determined to be unavailable.

There are the two types of frame loss measurement, defined by the ITU-T Y.1731 standards, ETH-LM and ETH-SLM. Junos OS supports only single-ended ETH-SLM. In single-ended ETH-SLM, each MEP sends frames with the ETH-SLM request information to its peer MEP and receives frames with ETH-SLM reply information from its peer MEP to perform synthetic loss measurements. Single-ended ETH-SLM is used for proactive or on-demand OAM to perform synthetic loss measurements applicable to point-to-point Ethernet connection. This method allows a MEP to initiate and report far-end and near-end loss measurements associated with a pair of MEPs that are part of the same maintenance entity group (MEG).

NOTE: MX Series Virtual Chassis does not support Ethernet synthetic loss measurement (ETH-SLM).

Single-ended ETH-SLM is used to perform on-demand or proactive tests by initiating a finite amount of ETH-SLM frames to one or multiple MEP peers and receiving the ETH-SLM reply from the peers. The ETH-SLM frames contain the ETH-SLM information that is used to measure and report both near-end and far-end synthetic loss measurements. Service-level agreement (SLA) measurement is the process of monitoring the bandwidth, delay, delay variation (*jitter*), continuity, and availability of a service. It enables you to identify network problems before customers are impacted by network defects. In proactive mode, SLA measurements are triggered by an iterator application. An iterator is designed to periodically transmit SLA measurement packets in the form of ITU-Y.1731-compliant frames for synthetic frame loss measurement . This mode differs from on-demand SLA measurement, which is user initiated. In on-demand mode, the measurements are triggered by the user through the CLI. When the user triggers the ETH-SLM through the CLI, the SLM request that is generated is as per the frame formats specified by the ITU-T Y.1731 standard.

NOTE: ACX5048 and ACX5096 routers support ETH-SLM for Layer 2 services.

Scenarios for Configuration of ETH-SLM

IN THIS SECTION

- Upstream MEP in MPLS Tunnels | **218**
- Downstream MEP in Ethernet Networks | **218**

ETH-SLM measures near-end and far-end frame loss between two MEPs that are part of the same MEG level. You can configure ETH-SLM to measure synthetic loss for both upward-facing or upstream MEP and downward-facing or downstream MEP. This section describes the following scenarios for the operation of ETH-SLM:

Upstream MEP in MPLS Tunnels

Consider a scenario in which a MEP is configured between the user network interfaces (UNIs) of two MX Series routers, MX1 and MX2, in the upstream direction. MX1 and MX2 are connected over an MPLS core network. ETH-SLM measurements are performed between the upstream MEP in the path linking the two routers. Both MX1 and MX2 can initiate on-demand or proactive ETH-SLM, which can measure both far-end and near-end loss at MX1 and MX2, respectively. The two UNIs are connected using MPLS-based Layer 2 VPN *virtual private wire service* (*VPWS*).

Downstream MEP in Ethernet Networks

Consider a scenario in which a MEP is configured between two MX Series routers, MX1 and MX2, on the Ethernet interfaces in the downstream direction. MX1 and MX2 are connected in an Ethernet topology and downstream MEP is configured toward the Ethernet network. ETH-SLM measurements are performed between the downstream MEP in the path linking the two routers. ETH-SLM can be measured in the path between these two routers.

Consider another scenario in which a MEP is configured in the downstream direction and service protection for a VPWS over MPLS is enabled by specifying a working path or protect path on the MEP. Service protection provides end-to-end connection protection of the working path in the event of a failure. To configure service protection, you must create two separate transport paths—a working path and a protect path. You can specify the working path and protect path by creating two maintenance

associations. To associate the maintenance association with a path, you must configure the MEP interface in the maintenance association and specify the path as working or protect.

In a sample topology, an MX Series router, MX1, is connected to two other MX Series routers, MX2 and MX3, over an MPLS core. The connectivity fault management (CFM) session between MX1 and MX2 is the working path on the MEP and the CFM session between MX1 and MX3 is the protect path on the MEP. MX2 and MX3 are, in turn, connected on Ethernet interfaces to MX4 in the access network. Downstream MEP is configured between MX1 and MX4 that passes through MX2 (working CFM session) and also between MX1 and MX4 that passes through MX3 (protected CFM session). ETH-SLM is performed between these downstream MEPs. In both the downstream MEPs, the configuration is performed on MX1 and MX4 UNIs, similar to upstream MEP.

Format of ETH-SLM Messages

IN THIS SECTION

- SLM PDU Format | **219**
- SLR PDU Format | **220**
- Data Iterator TLV Format | **221**

Synthetic loss messages (SLMs) support single-ended Ethernet synthetic loss measurement (ETH-SLM) requests. This topic contains the following sections that describe the formats of the SLM protocol data units (PDUs), SLR PDUs, and the data iterator type length value (TLV).

SLM PDU Format

The SLM PDU format is used by a MEP to transmit SLM information. The following components are contained in SLM PDUs:

- Source MEP ID—Source MEP ID is a 2-octet field where the last 13 least significant bits are used to identify the MEP transmitting the SLM frame. MEP ID is unique within the MEG.
- Test ID—Test ID is a 4-octet field set by the transmitting MEP and is used to identify a test when multiple tests run simultaneously between MEPs (including both concurrent on-demand and proactive tests).
- TxFCf—TxFCf is a 4-octet field that carries the number of SLM frames transmitted by the MEP toward its peer MEP.

The following are the fields in an SLM PDU:

- MEG Level—Configured maintenance domain level in the range 0–7.
- Version—0.
- OpCode—Identifies an OAM PDU type. For SLM, it is 55.
- Flags—Set to all zeros.
- TLV Offset—16.
- Source MEP ID—A 2-octet field used to identify the MEP transmitting the SLM frame. In this 2-octet field, the last 13 least significant bits are used to identify the MEP transmitting the SLM frame. MEP ID is unique within the MEG.
- RESV—Reserved fields are set to all zeros.
- Test ID—A 4-octet field set by the transmitting MEP and used to identify a test when multiple tests run simultaneously between MEPs (including both concurrent on-demand and proactive tests).
- TxFCf—A 4-octet field that carries the number of SLM frames transmitted by the MEP toward its peer MEP.
- Optional TLV—A data TLV may be included in any SLM transmitted. For the purpose of ETH-SLM, the value part of data TLV is unspecified.
- End TLV—All zeros octet value.

SLR PDU Format

The synthetic loss reply (SLR) PDU format is used by a MEP to transmit SLR information. The following are the fields in an SLR PDU:

- MEG Level—A 3-bit field the value of which is copied from the last received SLM PDU.
- Version—A 5-bit field the value of which is copied from the last received SLM PDU.
- OpCode—Identifies an OAM PDU type. For SLR, it is set as 54.
- Flags—A 1-octet field copied from the SLM PDU.
- TLV Offset—A 1-octet field copied from the SLM PDU.
- Source MEP ID—A 2-octet field copied from the SLM PDU.
- Responder MEP ID—A 2-octet field used to identify the MEP transmitting the SLR frame.
- Test ID—A 4-octet field copied from the SLM PDU.
- TxFCf—A 4-octet field copied from the SLM PDU.

- TxFCb—A 4 octet field. This value represents the number of SLR frames transmitted for this test ID.
- Optional TLV—The value is copied from the SLM PDU, if present.
- End TLV—A 1-octet field copied from the SLM PDU.

Data Iterator TLV Format

The data iterator TLV specifies the data TLV portion of the Y.1731 data frame. The MEP uses a data TLV when the MEP is configured to measure delay and delay variation for different frame sizes. The following are the fields in a data TLV:

- Type—Identifies the TLV type; value for this TLV type is Data (3).
- Length—Identifies the size, in octets, of the Value field containing the data pattern. The maximum value of the Length field is 1440.
- Data pattern—An *n*-octet (*n* denotes length) arbitrary bit pattern. The receiver ignores it.

Transmission of ETH-SLM Messages

IN THIS SECTION

- Initiation and Transmission of SLM Requests | **222**
- Reception of SLMs and Transmission of SLRs | **222**
- Reception of SLRs | **223**
- Computation of Frame Loss | **223**

The ETH-SLM functionality can process multiple synthetic loss message (SLM) requests simultaneously between a pair of MEPs. The session can be a proactive or an on-demand SLM session. Each SLM request is identified uniquely by a test ID.

A MEP can send SLM requests or respond to SLM requests. A response to an SLM request is called a synthetic loss reply (SLR). After a MEP determines an SLM request by using the test ID, the MEP calculates the far-end and near-end frame loss on the basis of the information in the SLM message or the SLM protocol data unit (PDU).

A MEP maintains the following local counters for each test ID and for each peer MEP being monitored in a maintenance entity for which loss measurements are to be performed:

- TxFCl—Number of synthetic frames transmitted toward the peer MEP for a test ID. A source MEP increments this number for successive transmission of synthetic frames with ETH-SLM request

information while a destination or receiving MEP increments this value for successive transmission of synthetic frames with the SLR information.

- RxFCl—Number of synthetic frames received from the peer MEP for a test ID. A source MEP increments this number for successive reception of synthetic frames with SLR information while a destination or receiving MEP increments it for successive reception of synthetic frames with ETH-SLM request information.

The following sections describe the phases of processing of SLM PDUs to determine synthetic frame loss:

Initiation and Transmission of SLM Requests

A MEP periodically transmits an SLM request with the OpCode field set as 55. The MEP generates a unique Test ID for the session, adds the source MEP ID, and initializes the local counters for the session before SLM initiation. For each SLM PDU transmitted for the session (test ID), the local counter TxFCl is sent in the packet.

No synchronization is required of the test ID value between initiating and responding MEPs because the test ID is configured at the initiating MEP, and the responding MEP uses the test ID it receives from the initiating MEP. Because ETH-SLM is a sampling technique, it is less precise than counting the service frames. Also, the accuracy of measurement depends on the number of SLM frames used or the period for transmitting SLM frames.

Reception of SLMs and Transmission of SLRs

After the destination MEP receives a valid SLM frame from the source MEP, an SLR frame is generated and transmitted to the requesting or source MEP. The SLR frame is valid if the MEG level and the destination MAC address match the receiving MEP's MAC address. All the fields in the SLM PDUs are copied from the SLM request except for the following fields:

- The source MAC address is copied to the destination MAC address and the source address contains the MEP's MAC address.
- The value of the OpCode field is changed from SLM to SLR (54).
- The responder MEP ID is populated with the MEP's MEP ID.
- TxFCb is saved with the value of the local counter RxFCl at the time of SLR frame transmission.
- An SLR frame is generated every time an SLM frame is received; therefore, RxFCl in the responder is equal to the number of SLM frames received and also equal to the number of SLR frames sent. At the responder or receiving MEP, RxFCl equals TxFCl.

Reception of SLRs

After an SLM frame (with a given TxFCf value) is transmitted, a MEP expects to receive a corresponding SLR frame (carrying the same TxTCf value) within the timeout value from its peer MEP. SLR frames that are received after the timeout value (5 seconds) are discarded. With the information contained in SLR frames, a MEP determines the frame loss for the specified measurement period. The measurement period is a time interval during which the number of SLM frames transmitted is statistically adequate to make a measurement at a given accuracy. A MEP uses the following values to determine near-end and far-end frame loss during the measurement period:

- Last received SLR frame's TxFCf and TxFCb values and the local counter RxFCl value at the end of the measurement period. These values are represented as TxFCf[tc], TxFCb[tc], and RxFCl[tc], where tc is the end time of the measurement period.
- SLR frame's TxFCf and TxFCb values of the first received SLR frame after the test starts and local counter RxFCl at the beginning of the measurement period. These values are represented as TxFCf[tp], TxFCb[tp], and RxFCl[tp], where tp is the start time of the measurement period.

For each SLR packet that is received, the local RxFCl counter is incremented at the sending or source MEP.

Computation of Frame Loss

Synthetic frame loss is calculated at the end of the measurement period on the basis of the value of the local counters and the information from the last frame received. The last received frames contains the TxFCf and TxFCb values. The local counter contains the RxFCl value. Using these values, frame loss is determined using the following formula:

Frame loss (far-end) = TxFCf – TxFCb

Frame loss (near-end) = TxFCb – RxFCl

Release History Table

Release	Description
16.1	Starting Junos OS Release 16.1, the Ethernet loss measurement (ETH-LM) results are inaccurate when connectivity fault management (CFM) and performance monitoring (PM) PDUs received locally at a maintenance endpoint (MEP) as classified as belonging to the yellow class or a packet loss priority (PLP) of medium-high.
16.1	Starting with Junos OS Release 16.1, performance monitoring for connectivity fault management (by including the `performance-monitoring` statement and its substatements at the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level) is not supported when the network-to-network (NNI) or egress interface is an aggregated Ethernet interface with member links on DPCs.

Configure Ethernet Frame Delay Measurement Sessions

IN THIS SECTION

- Guidelines for Configuring Routers to Support an ETH-DM Session | 224
- Guidelines for Starting an ETH-DM Session | 226
- Guidelines for Managing ETH-DM Statistics and ETH-DM Frame Counts | 228
- Configuring Routers to Support an ETH-DM Session | 234
- Triggering an Ethernet Frame Delay Measurements Session | 239
- Starting an ETH-DM Session | 241
- Example: Configuring One-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 244
- Example: Configuring Two-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 250
- Managing Continuity Measurement Statistics | 256
- Viewing Ethernet Frame Delay Measurements Statistics | 258
- Managing ETH-DM Statistics and ETH-DM Frame Counts | 259

Use this topic to understand how to configure Ethernet frame delay measurement sessions. You can start either a one-way Ethernet delay measurement session or a two-way Ethernet delay measurement session. Also, use this topic to view the delay measurement statistics and frame counts.

Guidelines for Configuring Routers to Support an ETH-DM Session

IN THIS SECTION

- Configuration Requirements for ETH-DM | 224
- Configuration Options for ETH-DM | 225

Keep the following guidelines in mind when configuring routers to support an Ethernet frame delay measurement (ETH-DM) session:

Configuration Requirements for ETH-DM

You can obtain ETH-DM information for a link that meets the following requirements:

- The measurements can be performed between peer maintenance association endpoints (MEPs) on two routers.
- The two MEPs must be configured on two Ethernet physical interfaces or on two Ethernet logical interfaces. For more information, see "Configuring a MEP to Generate and Respond to CFM Protocol Messages" on page 35.
- The two MEPs must be configured—on their respective routers—under the same maintenance association (MA) identifier. For more information, see "Creating a Maintenance Association" on page 28.
- On both routers, the MA must be associated with the same maintenance domain (MD) name. For more information, see "Creating a Maintenance Domain" on page 27.
- On both routers, periodic packet management (PPM) must be running on the Routing Engine and Packet Forwarding Engine, which is the default configuration. You can disable PPM on the Packet Forwarding Engine only. However, the Ethernet frame delay measurement feature requires that distributed PPM remain enabled on the Packet Forwarding Engine of both routers. For more information about `ppm`, see the Junos OS Routing Protocols Library for Routing Devices.
- If the PPM process (`ppm`) is disabled on the Packet Forwarding Engine, you must re-enable it. Re-enabling distributed `ppm` entails restarting the `ethernet-connectivity-fault-management` process, which causes all connectivity fault management (CFM) sessions to re-establish. For more information about CFM sessions, see *Configuring Ethernet Local Management Interface*.

NOTE: The Ethernet frame delay measurement feature is supported only for MEPs configured on Ethernet physical or logical interfaces on DPCs in MX Series routers. The ETH-DM feature is not supported on aggregated Ethernet interfaces or LSI pseudowires.

Configuration Options for ETH-DM

By default, the ETH-DM feature calculates frame delays using software-based timestamping of the ETH-DM PDU frames sent and received by the MEPs in the session. As an option that can increase the accuracy of ETH-DM calculations when the DPC is loaded with heavy traffic in the receive direction, you can enable hardware-assisted timestamping of session frames in the receive direction.

SEE ALSO

Ethernet Alarm Indication | 341

Inline Transmission Mode | 356

Guidelines for Starting an ETH-DM Session

IN THIS SECTION

- ETH-DM Session Prerequisites | 226
- ETH-DM Session Parameters | 226
- Restrictions for an ETH-DM Session | 227

Keep the following guidelines in mind when preparing to start an Ethernet frame delay measurement (ETH-DM) session:

ETH-DM Session Prerequisites

Before you can start an ETH-DM session, you must configure two MX Series routers to support ETH-DM by defining the two CFM-enabled physical or logical Ethernet interfaces on each router. This entails creating and configuring CFM maintenance domains, maintenance associations, and maintenance association end points on each router. For more information about enabling CFM on an Ethernet interface, see "Creating a Maintenance Domain" on page 27.

> **NOTE**: The Ethernet frame delay measurement feature is supported only for maintenance association end points configured on Ethernet physical or logical interfaces on DPCs in MX Series routers. The ETH-DM feature is not supported on aggregated Ethernet interfaces or LSI pseudowires.

For specific information about configuring routers to support ETH-DM, see"Guidelines for Configuring Routers to Support an ETH-DM Session" on page 224 and "Configuring Routers to Support an ETH-DM Session" on page 234.

ETH-DM Session Parameters

You can initiate a one-way or two-way ETH-DM session by entering the `monitor ethernet delay-measurement` operational command at a router that contains one end of the service for which you want to measure frame delay. The command options specify the ETH-DM session in terms of the CFM elements:

- The type of ETH-DM measurement (one-way or two-way) to be performed.
- The Ethernet service for which the ETH-DM measurement is to be performed:

- CFM maintenance domain—Name of the existing maintenance domain (MD) for which you want to measure Ethernet frame delays. For more information, see "Creating a Maintenance Domain" on page 27.

- CFM maintenance association—Name of an existing maintenance association (MA) within the maintenance domain. For more information, see "Creating a Maintenance Association" on page 28.

- Remote CFM maintenance association end point—The unicast MAC address or the numeric identifier of the remote maintenance association end point (MEP)—the physical or *logical interface* on the remote router that resides in the specified MD and is named in the specified MA—with which to perform the ETH-DM session. For more information, see "Configuring a MEP to Generate and Respond to CFM Protocol Messages" on page 35.

- Optional specifications:

  - Count—You can specify the number of ETH-DM requests to send for this frame delay measurement session. The range is from 1 through 65,535 frames. The default value is 10 frames.

    NOTE: Although you can trigger frame delay collection for up to 65,535 ETH-DM requests at a time, a router stores only the last 100 frame delay statistics per CFM session (pair of peer MEPs).

  - Frame interval—You can specify the number of seconds to elapse between ETH-DM frame transmittals. The default value is 1 second.

For more detailed information about the parameters you can specify to start an ETH-DM session, see the `monitor ethernet delay-measurement` operational command description in the CLI Explorer.

Restrictions for an ETH-DM Session

The following restrictions apply to an ETH-DM session:

- You cannot run multiple simultaneous ETH-DM sessions with the same remote MEP or MAC address.

- For a given ETH-DM session, you can collect frame delay information for a maximum of 65,535 frames.

- For a given CFM session (pair of peer MEPs), the ETH-DM database stores a maximum of 100 statistics, with the older statistics being “aged out” as newer statistics are collected for that pair of MEPs.

  - For one-way delay measurements collected within the same CFM session, the 100 most recent ETH-DM statistics can be retrieved at any point of time at the router on which the receiver MEP is defined.

- For two-way delay measurements collected within the same CFM session, the 100 most recent ETH-DM statistics can be retrieved at any point of time at the router on which the initiator MEP is defined.

  Depending on the number of frames exchanged in the individual ETH-DM sessions, the ETH-DM database can contain statistics collected through multiple ETH-DM sessions.

- If *graceful Routing Engine switchover* (GRES) occurs, any collected ETH-DM statistics are lost, and ETH-DM frame counts are reset to zeroes. GRES enables a router with dual Routing Engines to switch from a primary Routing Engine to a backup Routing Engine without interruption to packet forwarding. For more information, see the Junos OS High Availability User Guide.

- Accuracy of frame delay data is compromised when the system is changing (such as from reconfiguration). We recommend performing Ethernet frame delay measurements on a stable system.

SEE ALSO

monitor ethernet delay-measurement

Guidelines for Managing ETH-DM Statistics and ETH-DM Frame Counts

IN THIS SECTION

- ETH-DM Statistics | **228**
- ETH-DM Statistics Retrieval | **231**
- ETH-DM Frame Counts | **231**
- ETH-DM Frame Count Retrieval | **232**

ETH-DM Statistics

Ethernet frame delay statistics are the frame delay and frame delay variation values determined by the exchange of frames containing ETH-DM protocol data units (PDUs).

- For a one-way ETH-DM session, statistics are collected in an ETH-DM database at the router that contains the receiver MEP. For a detailed description of one-way Ethernet frame delay measurement, including the exchange of one-way delay PDU frames, see "Ethernet Frame Delay Measurements Overview" on page 205.

- For a two-way ETH-DM session, statistics are collected in an ETH-DM database at the router that contains the initiator MEP. For a detailed description of two-way Ethernet frame delay measurement,

including the exchange of two-way delay PDU frames, see "Ethernet Frame Delay Measurements Overview" on page 205.

A CFM database stores CFM-related statistics and—for Ethernet interfaces that support ETH-DM—the 100 most recently collected ETH-DM statistics for that pair of MEPs. You can view ETH-DM statistics by using the `delay-statistics` or `mep-statistics` form of the `show oam ethernet connectivity-fault-management` command to display the CFM statistics for the MEP that collects the ETH-DM statistics you want to view.

Table 13 on page 229 describes the ETH-DM statistics calculated in an ETH-DM session.

Table 13: ETH-DM Statistics

Field Name	Field Description
`One-way delay (µsec)`†	For a one-way ETH-DM session, the frame delay, in microseconds, collected at the receiver MEP. To display frame delay statistics for a given one-way ETH-DM session, use the `delay-statistics` or `mep-statistics` form of the `show oam ethernet connectivity-fault-management` command at the receiver MEP for that session.
`Two-way delay (µsec)`	For a two-way ETH-DM session, the frame delay, in microseconds, collected at the initiator MEP. When you start a two-way frame delay measurement, the CLI output displays each DMR frame receipt timestamp and corresponding DMM frame delay and delay variation collected as the session progresses. To display frame delay statistics for a given two-way ETH-DM session, use the `delay-statistics` or `mep-statistics` form of the `show oam ethernet connectivity-fault-management` command at the initiator MEP for that session.

Table 13: ETH-DM Statistics *(Continued)*

Field Name	Field Description
`Average delay`†	When you start a two-way frame delay measurement, the CLI output includes a runtime display of the average two-way frame delay among the statistics collected for the ETH-DM session only. When you display ETH-DM statistics using a `show` command, the `Average delay` field displays the average one-way and two- frame delays among all ETH-DM statistics collected at the CFM session level. For example, suppose you start two one-way ETH-DM sessions for 50 counts each, one after the other. If, after both measurement sessions complete, you use a `show` command to display 100 ETH-DM statistics for that CFM session, the `Average delay` field displays the average frame delay among all 100 statistics.
`Average delay variation`†	When you start a two-way frame delay measurement, the CLI output includes a runtime display of the average two-way frame delay variation among the statistics collected for the ETH-DM session only. When you display ETH-DM statistics using a `show` command, the `Average delay variation` field displays the average one-way and two- frame delay variations among all ETH-DM statistics collected at the CFM session level.
`Best-case delay`†	When you start a two-way frame delay measurement, the CLI output includes a runtime display of the lowest two-way frame delay value among the statistics collected for the ETH-DM session only. When you display ETH-DM statistics using a `show` command, the `Best case delay` field displays the lowest one-way and two-way frame delays among all ETH-DM statistics collected at the CFM session level.
`Worst-case delay`†	When you start a two-way frame delay measurement, the CLI output includes a runtime display of the highest two-way frame delay value among the statistics collected for the ETH-DM session only. When you display ETH-DM statistics using a `show` command, the `Worst case delay` field displays the highest one-way and two-way frame delays among all statistics collected at the CFM session level.

Table 13: ETH-DM Statistics *(Continued)*

Field Name	Field Description

†When you start a one-way frame delay measurement, the CLI output displays NA (“not available”) for this field. One-way ETH-DM statistics are collected at the remote (receiver) MEP. Statistics for a given one-way ETH-DM session are available only by displaying CFM statistics for the receiver MEP.

ETH-DM Statistics Retrieval

At the receiver MEP for a one-way session, or at the initiator MEP for a two-way session, you can display all ETH-DM statistics collected at a CFM session level by using the following operational commands:

- `show oam ethernet connectivity-fault-management delay-statistics maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* `<local-mep` *`mep-id`*`> <remote-mep` *`mep-id`*`> <count` *`count`*`>`
- `show oam ethernet connectivity-fault-management mep-statistics maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* `<local-mep` *`mep-id`*`> <remote-mep` *`mep-id`*`> <count` *`count`*`>`

ETH-DM Frame Counts

The number of ETH-DM PDU frames exchanged in a ETH-DM session are stored in the CFM database on each router.

Table 14 on page 231 describes the ETH-DM frame counts collected in an ETH-DM session.

Table 14: ETH-DM Frame Counts

Field Name	Field Description
`1DMs sent`	Number of one-way delay measurement (1DM) PDU frames sent to the peer MEP in this session. Stored in the CFM database of the MEP initiating a one-way frame delay measurement.
`Valid 1DMs received`	Number of valid 1DM frames received. Stored in the CFM database of the MEP receiving a one-way frame delay measurement.

Table 14: ETH-DM Frame Counts *(Continued)*

Field Name	Field Description
`Invalid 1DMs received`	Number of invalid 1DM frames received. Stored in the CFM database of the MEP receiving a one-way frame delay measurement.
`DMMs sent`	Number of delay measurement message (DMM) PDU frames sent to the peer MEP in this session. Stored in the CFM database of the MEP initiating a two-way frame delay measurement.
`DMRs sent`	Number of delay measurement reply (DMR) frames sent (in response to a received DMM). Stored in the CFM database of the MEP responding to a two-way frame delay measurement.
`Valid DMRs received`	Number of valid DMR frames received. Stored in the CFM database of the MEP initiating a two-way frame delay measurement.
`Invalid DMRs received`	Number of invalid DMR frames received. Stored in the CFM database of the MEP initiating a two-way frame delay measurement.

ETH-DM Frame Count Retrieval

Each router counts the number of ETH-DM frames sent or received and stores the counts in a CFM database.

Frame Counts Stored in CFM Databases

You can display ETH-DM frame counts for MEPs assigned to specified Ethernet interfaces or for specified MEPs in CFM sessions by using the following operational commands:

- `show oam ethernet connectivity-fault-management interfaces` (`detail` | `extensive`)
- `show oam ethernet connectivity-fault-management mep-database maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* <local-mep *`mep-id`*> <remote-mep *`mep-id`*>

One-Way ETH-DM Frame Counts

For a one-way ETH-DM session, delay statistics are collected at the receiver MEP only, but frame counts are collected at both MEPs. As indicated in Table 14 on page 231, one-way ETH-DM frame counts are tallied from the perspective of each router in the session:

- At the initiator MEP, the router counts the number of 1DM frames sent.
- At the receiver MEP, the router counts the number of valid 1DM frames received and the number of invalid 1DM frames received.

You can also view one-way ETH-DM frame counts—for a receiver MEP—by using the `show oam ethernet connectivity-fault-management mep-statistics` command to display one-way statistics and frame counts together.

Two-Way ETH-DM Frame Counts

For a two-way ETH-DM session, delay statistics are collected at the initiator MEP only, but frame counts are collected at both MEPs. As indicated in Table 14 on page 231, two-way ETH-DM frame counts are tallied from the perspective of each router in the session:

- At the initiator MEP, the router counts the number of DMM frames sent, valid DMR frames received, and invalid DMR frames received.
- At the responder MEP, the router counts the number of DMR frames sent.

You can also view two-way ETH-DM frame counts—for an initiator MEP—by using the `show oam ethernet connectivity-fault-management mep-statistics` command to display two-way statistics and frame counts together.

SEE ALSO

clear oam ethernet connectivity-fault-management statistics

show oam ethernet connectivity-fault-management mep-statistics

show oam ethernet connectivity-fault-management delay-statistics

show oam ethernet connectivity-fault-management interfaces

show oam ethernet connectivity-fault-management mep-database

Configuring Routers to Support an ETH-DM Session

IN THIS SECTION

- Configuring MEP Interfaces | 234
- Ensuring That Distributed ppm Is Not Disabled | 235
- Enabling the Hardware-Assisted Timestamping Option | 238
- Configuring the Server-Side Processing Option | 238

Configuring MEP Interfaces

Before you can start an Ethernet frame delay measurement session across an Ethernet service, you must configure two MX Series routers to support ETH-DM.

To configure an Ethernet interface on a MX Series router to support ETH-DM:

1. On each router, configure two physical or logical Ethernet interfaces connected by a VLAN. The following configuration is typical for single-tagged logical interfaces:

```
[edit interfaces]
interface {
    ethernet-interface-name {
        vlan-tagging;
        unit logical-unit-number {
            vlan-id vlan-id; # Both interfaces on this VLAN
        }
    }
}
```

   Both interfaces will use the same VLAN ID.

2. On each router, attach peer MEPs to the two interfaces. The following configuration is typical:

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain md-name { # On both routers
                level number;
```

```
                maintenance-association ma-name { # On both routers
                    continuity-check {
                        interval 100ms;
                        hold-interval 1;
                    }
                    mep mep-id { # Attach to VLAN interface
                        auto-discovery;
                        direction (up | down);
                        interface interface-name;
                        priority number;
                    }
                }
            }
        }
    }
}
```

Ensuring That Distributed ppm Is Not Disabled

By default, the router's period packet management process (ppm) runs sessions distributed to the Packet Forwarding Engine in addition to the Routing Engine. This process is responsible for periodic transmission of packets on behalf of its various client processes, such as Bidirectional Forwarding Detection (BFD), and it also receives packets on behalf of client processes.

In addition, ppm handles time-sensitive periodic processing and performs such processes as sending process-specific packets and gathering statistics. With ppm processes running distributed on both the Routing Engine and the Packet Forwarding Engine, you can run such processes as BFD on the Packet Forwarding Engine.

Distributed ppm Required for ETH-DM

Ethernet frame delay measurement requires that ppm remains distributed to the Packet Forwarding Engine. If ppm is not distributed to the Packet Forwarding Engines of both routers, ETH-DM PDU frame timestamps and ETH-DM statistics are not valid.

Before you start ETH-DM, you must verify that the following configuration statement is *NOT* present:

```
[edit]
routing-options {
     ppm {
        no-delegate-processing;
```

```
    }
}
```

If distributed ppm processing is disabled (as shown in the stanza above) on either router, you must re-enable it in order to use the ETH-DM feature.

Procedure to Ensure that Distributed ppm is Not Disabled

To ensure that distributed ppm is not disabled on a router:

1. Display the packet processing management (PPM) configuration to determine whether distributed ppm is disabled.
   - In the following example, distributed ppm is enabled on the router. In this case, you do not need to modify the router configuration:

     ```
     [edit]
     user@host# show routing-options
     ppm;
     ```

   - In the following example, distributed ppm is disabled on the router. In this case, you must proceed to Step 2 to modify the router configuration:

     ```
     [edit]
     user@host show routing-options
     ppm {
         no-delegate-processing;
     }
     ```

2. Modify the router configuration to re-enable distributed ppm and restart the Ethernet OAM Connectivity Fault Management process *ONLY IF* distributed ppm is disabled (as determined in the previous step).
   a. Before continuing, make any necessary preparations for the possible loss of connectivity on the router.

      Restarting the ethernet-connectivity-fault-management process has the following effect on your network:

      - All connectivity fault management (CFM) sessions re-establish.
      - All ETH-DM requests on the router terminate.

- All ETH-DM statistics and frame counts reset to 0.

b. Modify the router configuration to re-enable distributed `ppm`. For example:

```
[edit]
user@host# delete routing-options ppm no-delegate-processing
```

c. Commit the updated router configuration. For example:

```
[edit]
user@host# commit and-quit
commit complete
exiting configuration mode
```

d. To restart the Ethernet OAM Connectivity-Fault-Management process, enter the `restart ethernet-connectivity-fault-management <gracefully | immediately | soft>` operational mode command. For example:

```
user@host> restart ethernet-connectivity-fault-management
Connectivity fault management process started, pid 9893
```

Connectivity fault management (CFM) sessions operate in centralized mode over AE interfaces by default. Y.1731 performance monitoring (PM) is supported on centralized CFM sessions over AE interfaces. Also, distribution of CFM session over AE interfaces to line cards is supported from Junos OS Release 13.3. To enable the distribution of CFM sessions and to operate in centralized mode, include the `ppm delegate-processing` statement at the `[edit routing-options ppm]` hierarchy level. The mechanism that enables distribution of CFM sessions over AE interfaces provides the underlying infrastructure to support PM over AE interfaces. In addition, periodic packet management (PPM) handles time-sensitive periodic processing and performs such processes as sending process-specific packets and gathering statistics. With PPM processes running distributed on both the Routing Engine and the Packet Forwarding Engine, you can run performance monitoring processes on the Packet Forwarding Engine.

SEE ALSO

Periodic Packet Management

Understanding Periodic Packet Management on MX Series Routers

Enabling the Hardware-Assisted Timestamping Option

By default, Ethernet frame delay measurement uses software for timestamping transmitted and received ETH-DM frames. For Ethernet interfaces, you can optionally use hardware timing to assist in the timestamping of received ETH-DM frames to increase the accuracy of delay measurements.

Enabling hardware-assisted timestamping of received frames can increase the accuracy of ETH-DM calculations when the DPC is loaded with heavy traffic in the receive direction.

Starting in Junos OS Release 20.4R1, by default the hardware assistance is used for timestamping Ethernet frame delay frames on AFT based MX Series line cards, even if the `hardware-assisted-timestamping` is not configured.

To enable Ethernet frame delay measurement hardware assistance on the reception path, include the `hardware-assisted-timestamping` statement at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring]` hierarchy level:

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management {
            performance-monitoring {
                hardware-assisted-timestamping;
            }
        }
    }
}
```

SEE ALSO

hardware-assisted-timestamping

Configuring the Server-Side Processing Option

You can delegate the server-side processing (for both two-way delay measurement and loss measurement) to the Packet Forwarding Engine to prevent overloading on the Routing Engine. By default, the server-side processing is done by the Routing Engine.

To configure the server-side processing option:

1. In configuration mode, go to the following hierarchy level:

```
user@host# edit protocols oam ethernet connectivity-fault-management performance-monitoring
```

2. Configure the server-side processing option.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# set delegate-server-processing
```

3. Verify the configuration.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# show
performance-monitoring {
    delegate-server-processing;
}
```

SEE ALSO

> *delegate-server-processing*

RELATED DOCUMENTATION

> *Understanding Periodic Packet Management on MX Series Routers*
>
> Inline Transmission Mode | 356

Triggering an Ethernet Frame Delay Measurements Session

Before Ethernet frame delay measurement statistics can be displayed, they must be collected. To trigger Ethernet frame delay measurement, use the `monitor ethernet delay-measurement (one-way | two-way) (`*remote-mac-address*`) maintenance-domain` *name* `maintenance-association` *ma-id* `[count` *count*`] [wait` *time*`]` operational command.

The fields for this command are described in Table 15 on page 240.

Table 15: Monitor Ethernet Delay Command Parameters

Parameter	Parameter Range	Description
`one-way` or `two-way` or	NA	Perform a one-way or two-way (round-trip) delay measurement.
remote-mac-address	Unicast MAC address	Send delay measurement frames to the destination unicast MAC address (use the format xx:xx:xx:xx:xx:xx). Multicast MAC addresses are not supported.
`mep` *identifier*	1–8191	The MEP identifier to use for the measurement. The discovered MAC address for this MEP identifier is used.
`maintenance-domain` *name*	Existing MD name	Specifies an existing maintenance domain (MD) to use for the measurement.
`maintenance-association` *ma-id*	Existing MA identifier	Specifies an existing maintenance association (MA) identifier to use for the measurement.
`count` *count*	1–65535 (default: 10)	(Optional) Specifies the number of Ethernet frame delay frames to send. The default is 10.
wait *time*	1–255 seconds (default: 1)	(Optional) Specifies the number of seconds to wait between frames. The default is 1 second.

If you attempt to monitor delays to a nonexistent MAC address, you must exit the application manually using **^C**:

```
user@host> monitor ethernet delay-measurement two-way 00:11:22:33:44:55
Two-way ETH-DM request to 00:11:22:33:44:55, Interface ge-5/2/9.0
^C
--- Delay measurement statistics ---
Packets transmitted: 10, Valid packets received: 0
Average delay: 0 usec, Average delay variation: 0 usec
Best case delay: 0 usec, Worst case delay: 0 usec
```

SEE ALSO

Configure Ethernet Frame Loss Measurement | 268

Starting an ETH-DM Session

IN THIS SECTION

- Using the monitor ethernet delay-measurement Command | 241
- Starting a One-Way ETH-DM Session | 242
- Starting a Two-Way ETH-DM Session | 243

Using the monitor ethernet delay-measurement Command

After you have configured two MX Series routers to support ITU-T Y.1731 Ethernet frame delay measurement (ETH-DM), you can initiate a one-way or two-way Ethernet frame delay measurement session from the CFM maintenance association end point (MEP) on one of the routers to the peer MEP on the other router.

To start an ETH-DM session between the specified local MEP and the specified remote MEP, enter the `monitor ethernet delay-measurement` command at operational mode. The syntax of the command is as follows:

```
monitor ethernet delay-measurement
(one-way | two-way)
maintenance-domain md-name
maintenance-association ma-name
(remote-mac-address | mep remote-mep-id)
<count frame-count>
<wait interval-seconds>
<priority 802.1p value>
<size>
<no-session-id-tlv>
<xml>
```

For a one-way frame delay measurement, the command displays a runtime display of the number of 1DM frames sent from the initiator MEP during that ETH-DM session. One-way frame delay and frame delay variation measurements from an ETH-DM session are collected in a CFM database at the router that contains the receiver MEP. You can retrieve ETH-DM statistics from a CFM database at a later time.

For a two-way frame delay measurement, the command displays two-way frame delay and frame delay variation values for each round-trip frame exchange during that ETH-DM session, as well as a runtime display of useful summary information about the session: average delay, average delay variation, best-case delay, and worst-case delay. Two-way frame delay and frame delay variation values measurements from an ETH-DM session are collected in a CFM database at the router that contains the initiator MEP. You can retrieve ETH-DM statistics from a CFM database at a later time.

NOTE: Although you can trigger frame delay collection for up to 65,535 ETH-DM requests at a time, a router stores only the last 100 frame delay statistics per CFM session (pair of peer MEPs).

For a complete description of the `monitor ethernet delay-measurement` operational command, see the CLI Explorer.

SEE ALSO

monitor ethernet delay-measurement

Starting a One-Way ETH-DM Session

To start a one-way Ethernet frame delay measurement session, enter the `monitor ethernet delay-measurement one-way` command from operational mode, and specify the peer MEP by its MAC address or by its MEP identifier.

For example:

```
user@host> monitor ethernet delay-measurement one-way 00:05:85:73:39:4a maintenance-domain md6
maintenance-association ma6 count 10
One-way ETH-DM request to 00:05:85:73:39:4a, Interface xe-5/0/0.0
1DM Frames sent : 10
--- Delay measurement statistics ---
Packets transmitted: 10
Average delay: NA, Average delay variation: NA
Best case delay: NA, Worst case delay: NA
```

NOTE: If you attempt to monitor delays to a nonexistent MAC address, you must type **Ctrl** + **C** to explicitly quit the `monitor ethernet delay-measurement` command and return to the CLI command prompt.

SEE ALSO

monitor ethernet delay-measurement

Starting a Two-Way ETH-DM Session

To start a two-way Ethernet frame delay measurement session, enter the `monitor ethernet delay-measurement two-way` command from operational mode, and specify the peer MEP by its MAC address or by its MEP identifier.

For example:

```
user@host> monitor ethernet delay-measurement two-way 00:05:85:73:39:4a maintenance-domain md6
maintenance-association ma6 count 10
Two-way ETH-DM request to 00:05:85:73:39:4a, Interface xe-5/0/0.0
DMR received from 00:05:85:73:39:4a Delay: 100 usec Delay variation: 0 usec
DMR received from 00:05:85:73:39:4a Delay: 92 usec  Delay variation: 8 usec
DMR received from 00:05:85:73:39:4a Delay: 92 usec  Delay variation: 0 usec
DMR received from 00:05:85:73:39:4a Delay: 111 usec Delay variation: 19 usec
DMR received from 00:05:85:73:39:4a Delay: 110 usec Delay variation: 1 usec
DMR received from 00:05:85:73:39:4a Delay: 119 usec Delay variation: 9 usec
DMR received from 00:05:85:73:39:4a Delay: 122 usec Delay variation: 3 usec
DMR received from 00:05:85:73:39:4a Delay: 92 usec  Delay variation: 30 usec
DMR received from 00:05:85:73:39:4a Delay: 92 usec  Delay variation: 0 usec
DMR received from 00:05:85:73:39:4a Delay: 108 usec Delay variation: 16 usec

--- Delay measurement statistics ---
Packets transmitted: 10, Valid packets received: 10
Average delay: 103 usec, Average delay variation: 8 usec
Best case delay: 92 usec, Worst case delay: 122 usec
```

NOTE: If you attempt to monitor delays to a nonexistent MAC address, you must type **Ctrl** + **C** to explicitly quit the `monitor ethernet delay-measurement` command and return to the CLI command prompt.

SEE ALSO

monitor ethernet delay-measurement

RELATED DOCUMENTATION

Inline Transmission Mode | 356

Example: Configuring One-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces

This example uses two MX Series routers: **MX-1** and **MX-2**. The configuration creates a CFM down MEP session on a VLAN-tagged logical interface connecting the two (**ge-5/2/9** on Router **MX-1** and **ge-0/2/5** on Router **MX-2**).

NOTE: These are not complete router configurations.

Configuration on Router **MX-1**:

```
[edit]
interfaces {
    ge-5/2/9 {
        vlan-tagging;
        unit 0 {
            vlan-id 512;
        }
    }
}
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                traceoptions {
                    file eoam_cfm.log size 1g files 2 world-readable;
                    flag all;
                }
                linktrace {
                    path-database-size 255;
                    age 10s;
                }
                maintenance-domain md6 {
                    level 6;
                    maintenance-association ma6 {
                        continuity-check {
                            interval 100ms;
```

```
                        hold-interval 1;
                    }
                    mep 201 {
                        interface ge-5/2/9.0;
                        direction down;
                        auto-discovery;
                    }
                }
            }
        }
    }
}
```

Configuration on Router **MX-2**:

```
[edit]
interfaces {
    ge-0/2/5 {
        vlan-tagging;
        unit 0 {
            vlan-id 512;
        }
    }
}
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                traceoptions {
                    file eoam_cfm.log size 1g files 2 world-readable;
                    flag all;
                }
                linktrace {
                    path-database-size 255;
                    age 10s;
                }
                maintenance-domain md6 {
                    level 6;
                    maintenance-association ma6 {
                        continuity-check {
                            interval 100ms;
```

```
                        hold-interval 1;
                    }
                    mep 101 {
                        interface ge-0/2/5.0;
                        direction down;
                        auto-discovery;
                    }
                }
            }
        }
    }
}
```

From Router **MX-2**, start a one-way delay measurement to Router **MX-1**.

```
user@MX-2> monitor ethernet delay-measurement one-way mep 201 maintenance-domain md6 maintenance-
association ma6 count 10
One-way ETH-DM request to 00:90:69:0a:43:94, Interface ge-0/2/5.0
1DM Frames sent : 10
--- Delay measurement statistics ---
Packets transmitted: 10
Average delay: NA, Average delay variation: NA
Best case delay: NA, Worst case delay: NA
```

The counters are displayed as part of the local MEP database on Router **MX-2**.

```
user@MX-2> show oam ethernet connectivity-fault-management mep-database maintenance-domain md6
maintenance-domain ma6
Maintenance domain name: md6, Format: string, Level: 6
  Maintenance association name: ma6, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 101, Direction: down, MAC address: 00:90:69:0a:48:57
  Auto-discovery: enabled, Priority: 0
  Interface name: ge-0/2/5.0, Interface status: Active, Link status: Up
  Defects:
    Remote MEP not receiving CCM                : no
    Erroneous CCM received                      : no
    Cross-connect CCM received                  : no
    RDI sent by some MEP                        : no
  Statistics:
```

```
    CCMs sent                                    : 1590
    CCMs received out of sequence                : 0
    LBMs sent                                    : 0
    Valid in-order LBRs received                 : 0
    Valid out-of-order LBRs received             : 0
    LBRs received with corrupted data            : 0
    LBRs sent                                    : 0
    LTMs sent                                    : 0
    LTMs received                                : 0
    LTRs sent                                    : 0
    LTRs received                                : 0
    Sequence number of next LTM request          : 0
    1DMs sent                                    : 10
    Valid 1DMs received                          : 0
    Invalid 1DMs received                        : 0
    DMMs sent                                    : 0
    DMRs sent                                    : 0
    Valid DMRs received                          : 0
    Invalid DMRs received                        : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
      201     00:90:69:0a:43:94       ok    ge-0/2/5.0
```

The remote MEP database statistics are available on Router **MX-1**.

```
user@MX-1> show oam ethernet connectivity-fault-management mep-database maintenance-domain md6
Maintenance domain name: md6, Format: string, Level: 6
  Maintenance association name: ma6, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 201, Direction: down, MAC address: 00:90:69:0a:43:94
  Auto-discovery: enabled, Priority: 0
  Interface name: ge-5/2/9.0, Interface status: Active, Link status: Up
  Defects:
    Remote MEP not receiving CCM                 : no
    Erroneous CCM received                       : no
    Cross-connect CCM received                   : no
    RDI sent by some MEP                         : no
  Statistics:
    CCMs sent                                    : 1572
    CCMs received out of sequence                : 0
    LBMs sent                                    : 0
    Valid in-order LBRs received                 : 0
```

```
    Valid out-of-order LBRs received              : 0
    LBRs received with corrupted data             : 0
    LBRs sent                                     : 0
    LTMs sent                                     : 0
    LTMs received                                 : 0
    LTRs sent                                     : 0
    LTRs received                                 : 0
    Sequence number of next LTM request           : 0
    1DMs sent                                     : 0
    Valid 1DMs received                           : 10
    Invalid 1DMs received                         : 0
    DMMs sent                                     : 0
    DMRs sent                                     : 0
    Valid DMRs received                           : 0
    Invalid DMRs received                         : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
      101     00:90:69:0a:48:57       ok    ge-5/2/9.0
```

The remote Router **MX-1** should also collect the delay statistics (up to 100 per session) for display with **mep-statistics** or **delay-statistics**.

```
user@MX-1> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain md6
MEP identifier: 201, MAC address: 00:90:69:0a:43:94
Remote MEP count: 1
  CCMs sent                                   : 3240
  CCMs received out of sequence               : 0
  LBMs sent                                   : 0
  Valid in-order LBRs received                : 0
  Valid out-of-order LBRs received            : 0
  LBRs received with corrupted data           : 0
  LBRs sent                                   : 0
  LTMs sent                                   : 0
  LTMs received                               : 0
  LTRs sent                                   : 0
  LTRs received                               : 0
  Sequence number of next LTM request         : 0
  1DMs sent                                   : 0
  Valid 1DMs received                         : 10
  Invalid 1DMs received                       : 0
  DMMs sent                                   : 0
  DMRs sent                                   : 0
```

```
  Valid DMRs received                           : 0
  Invalid DMRs received                         : 0

  Remote MEP identifier: 101
  Remote MAC address: 00:90:69:0a:48:57
    Delay measurement statistics:
    Index  One-way delay  Two-way delay
              (usec)         (usec)
      1        370
      2        357
      3        344
      4        332
      5        319
      6        306
      7        294
      8        281
      9        269
      10       255
    Average one-way delay          : 312 usec
    Average one-way delay variation: 11 usec
    Best case one-way delay        : 255 usec
    Worst case one-way delay       : 370 usec
```

```
user@MX-1> show oam ethernet connectivity-fault-management delay-statistics maintenance-domain
md6
MEP identifier: 201, MAC address: 00:90:69:0a:43:94
Remote MEP count: 1

  Remote MAC address: 00:90:69:0a:48:57
    Delay measurement statistics:
    Index  One-way delay  Two-way delay
              (usec)         (usec)
      1        370
      2        357
      3        344
      4        332
      5        319
      6        306
      7        294
      8        281
      9        269
```

```
      10       255
    Average one-way delay          : 312 usec
    Average one-way delay variation: 11 usec
    Best case one-way delay        : 255 usec
```

NOTE: When two systems are close to each other, their one-way delay values are very high compared to their two-way delay values. This is because one-way delay measurement requires the timing for the two systems to be synchronized at a very granular level and MX Series routers do not support this granular synchronization. However, two-way delay measurement does not require synchronized timing, making two-way delay measurements more accurate.

SEE ALSO

- Ethernet Interfaces User Guide for Routing Devices
- Ethernet Frame Delay Measurements Overview | **205**
- *Configuring MEP Interfaces to Support Ethernet Frame Delay Measurements*
- Triggering an Ethernet Frame Delay Measurements Session | **239**
- Viewing Ethernet Frame Delay Measurements Statistics | **258**

Example: Configuring Two-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces

This example uses two MX Series routers: **MX-1** and **MX-2**. The configuration creates a CFM down MEP session on a VLAN-tagged logical interface connecting the two (**ge-5/2/9** on Router **MX-1** and **ge-0/2/5** on Router **MX-2**).

NOTE: These are not complete router configurations.

Configuration on Router **MX-1**:

```
[edit]
interfaces {
    ge-5/2/9 {
        vlan-tagging;
        unit 0 {
            vlan-id 512;
```

```
        }
    }
}
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                traceoptions {
                    file eoam_cfm.log size 1g files 2 world-readable;
                    flag all;
                }
                linktrace {
                    path-database-size 255;
                    age 10s;
                }
                maintenance-domain md6 {
                    level 6;
                    maintenance-association ma6 {
                        continuity-check {
                            interval 100ms;
                            hold-interval 1;
                        }
                        mep 201 {
                            interface ge-5/2/9.0;
                            direction down;
                            auto-discovery;
                        }
                    }
                }
            }
        }
    }
}
```

Configuration on Router **MX-2**:

```
[edit]
interfaces {
    ge-0/2/5 {
        vlan-tagging;
        unit 0 {
            vlan-id 512;
```

```
        }
    }
}
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                traceoptions {
                    file eoam_cfm.log size 1g files 2 world-readable;
                    flag all;
                }
                linktrace {
                    path-database-size 255;
                    age 10s;
                }
                maintenance-domain md6 {
                    level 6;
                    maintenance-association ma6 {
                        continuity-check {
                            interval 100ms;
                            hold-interval 1;
                        }
                        mep 101 {
                            interface ge-0/2/5.0;
                            direction down;
                            auto-discovery;
                        }
                    }
                }
            }
        }
    }
}
```

From Router **MX-1**, start a two-way delay measurement to Router **MX-2**.

```
user@MX-1> monitor ethernet delay-measurement two-way mep 101 maintenance-domain md6 maintenance-
association ma6 count 10
Two-way ETH-DM request to 00:90:69:0a:48:57, Interface ge-5/2/9.0
DMR received from 00:90:69:0a:48:57 Delay: 100 usec Delay variation: 0 usec
DMR received from 00:90:69:0a:48:57 Delay: 92 usec Delay variation: 8 usec
DMR received from 00:90:69:0a:48:57 Delay: 92 usec Delay variation: 0 usec
```

```
DMR received from 00:90:69:0a:48:57 Delay: 111 usec Delay variation: 19 usec
DMR received from 00:90:69:0a:48:57 Delay: 110 usec Delay variation: 1 usec
DMR received from 00:90:69:0a:48:57 Delay: 119 usec Delay variation: 9 usec
DMR received from 00:90:69:0a:48:57 Delay: 122 usec Delay variation: 3 usec
DMR received from 00:90:69:0a:48:57 Delay: 92 usec Delay variation: 30 usec
DMR received from 00:90:69:0a:48:57 Delay: 92 usec Delay variation: 0 usec
DMR received from 00:90:69:0a:48:57 Delay: 108 usec Delay variation: 16 usec

--- Delay measurement statistics ---
Packets transmitted: 10, Valid packets received: 10
Average delay: 103 usec, Average delay variation: 8 usec
Best case delay: 92 usec, Worst case delay: 122 usec
```

The counters are displayed as part of the MEP database on Router **MX-1** maintenance domain **MD6**.

```
user@MX-1> show oam ethernet connectivity-fault-management mep-database maintenance-domain md6
Maintenance domain name: md6, Format: string, Level: 6
  Maintenance association name: ma6, Format: string
  Continuity-check status: enabled, Interval: 100ms, Loss-threshold: 3 frames
  MEP identifier: 201, Direction: down, MAC address: 00:90:69:0a:43:94
  Auto-discovery: enabled, Priority: 0
  Interface name: ge-5/2/9.0, Interface status: Active, Link status: Up
  Defects:
    Remote MEP not receiving CCM                : no
    Erroneous CCM received                      : no
    Cross-connect CCM received                  : no
    RDI sent by some MEP                        : no
  Statistics:
    CCMs sent                                   : 894
    CCMs received out of sequence               : 0
    LBMs sent                                   : 0
    Valid in-order LBRs received                : 0
    Valid out-of-order LBRs received            : 0
    LBRs received with corrupted data           : 0
    LBRs sent                                   : 0
    LTMs sent                                   : 0
    LTMs received                               : 0
    LTRs sent                                   : 0
    LTRs received                               : 0
    Sequence number of next LTM request         : 0
    1DMs sent                                   : 0
    Valid 1DMs received                         : 0
```

```
    Invalid 1DMs received                          : 0
    DMMs sent                                      : 10
    DMRs sent                                      : 0
    Valid DMRs received                            : 10
    Invalid DMRs received                          : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
      101     00:90:69:0a:48:57       ok    ge-5/2/9.0
```

The collected MEP statistics are saved (up to 100 per remote MEP or per CFM session) and displayed as part of the MEP statistics on Router **MX-1**.

```
user@MX-1> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain md6
MEP identifier: 201, MAC address: 00:90:69:0a:43:94
Remote MEP count: 1
  CCMs sent                                    : 3154
  CCMs received out of sequence                : 0
  LBMs sent                                    : 0
  Valid in-order LBRs received                 : 0
  Valid out-of-order LBRs received             : 0
  LBRs received with corrupted data            : 0
  LBRs sent                                    : 0
  LTMs sent                                    : 0
  LTMs received                                : 0
  LTRs sent                                    : 0
  LTRs received                                : 0
  Sequence number of next LTM request          : 0
  1DMs sent                                    : 0
  Valid 1DMs received                          : 0
  Invalid 1DMs received                        : 0
  DMMs sent                                    : 10
  DMRs sent                                    : 0
  Valid DMRs received                          : 10
  Invalid DMRs received                        : 0

  Remote MEP identifier: 101
  Remote MAC address: 00:90:69:0a:48:57
    Delay measurement statistics:
    Index  One-way delay  Two-way delay
              (usec)         (usec)
      1                      100
      2                      92
```

```
      3                       92
      4                       111
      5                       110
      6                       119
      7                       122
      8                       92
      9                       92
      10                      108
    Average two-way delay          : 103 usec
    Average two-way delay variation: 8 usec
    Best case two-way delay        : 92 usec
    Worst case two-way delay       : 122 usec
```

The collected delay statistics are also saved (up to 100 per session) and displayed as part of the MEP delay statistics on Router **MX-1**.

```
user@MX-1> show oam ethernet connectivity-fault-management delay-statistics maintenance-domain
md6
MEP identifier: 201, MAC address: 00:90:69:0a:43:94
Remote MEP count: 1

  Remote MAC address: 00:90:69:0a:48:57
    Delay measurement statistics:
    Index  One-way delay  Two-way delay
                          (usec)         (usec)
      1                      100
      2                      92
      3                      92
      4                      111
      5                      110
      6                      119
      7                      122
      8                      92
      9                      92
      10                     108
    Average two-way delay          : 103 usec
    Average two-way delay variation: 8 usec
    Best case two-way delay        : 92 usec
    Worst case two-way delay       : 122 usec
```

SEE ALSO

- Ethernet Interfaces User Guide for Routing Devices
- Ethernet Frame Delay Measurements Overview | 205
- *Configuring MEP Interfaces to Support Ethernet Frame Delay Measurements*
- Triggering an Ethernet Frame Delay Measurements Session | 239
- Example: Configuring One-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 244
- Configuring ETH-DM with Untagged Interfaces

Managing Continuity Measurement Statistics

IN THIS SECTION

- Displaying Continuity Measurement Statistics | 256
- Clearing Continuity Measurement Statistics | 257

Displaying Continuity Measurement Statistics

IN THIS SECTION

- Purpose | 256
- Action | 257

Purpose

Display continuity measurement.

The `show oam ethernet connectivity-fault-management delay-statistics maintenance-domain md1 maintenance-association ma1` command is enhanced to display continuity measurement statistics for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

Action

- To display the ETH-DM statistics collected for MEPs belonging to MA `ma1` and within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management delay-statistics maintenance-
domain md1 maintenance-association ma1
```

SEE ALSO

show oam ethernet connectivity-fault-management delay-statistics

Clearing Continuity Measurement Statistics

IN THIS SECTION

- Purpose | 257
- Action | 257

Purpose

Clear the continuity measurement statistics

By default, statistics are deleted for all MEPs attached to CFM-enabled interfaces on the router. However, you can filter the scope of the command by specifying an interface name.

Action

- To clear the continuity measurement statistics for all MEPs attached to CFM-enabled interfaces on the router:

```
user@host> clear oam ethernet connectivity-fault-management continuity-measurement
maintenance-domain md-name maintenance-association ma-name local-mep local-mep-id remote-mep
remote-mep-id
```

SEE ALSO

clear oam ethernet connectivity-fault-management continuity-measurement

RELATED DOCUMENTATION

clear oam ethernet connectivity-fault-management continuity-measurement

show oam ethernet connectivity-fault-management delay-statistics

Viewing Ethernet Frame Delay Measurements Statistics

Once Ethernet frame delay measurement statistics have been collected, they can be displayed.

To retrieve the last 100 Ethernet frame delay measurement statistics per remote MEP or per CFM session, two types of `show` commands are provided:

- For all OAM frame counters and Ethernet frame delay measurement statistics
- For Ethernet frame delay measurement statistics only

To retrieve all Ethernet frame delay measurement statistics for a given session, use the `show oam ethernet connectivity-fault-management mep-statistics maintenance-domain` *`name`* `maintenance-association` *`name`* `[local-mep` *`identifier`*`] [remote-mep` *`identifier`*`] [count` *`count`*`]` command.

To retrieve only Ethernet frame delay measurement statistics for a given session, use the `show oam ethernet connectivity-fault-management delay-statistics maintenance-domain` *`name`* `maintenance-association` *`name`* `[local-mep` *`identifier`*`] [remote-mep` *`identifier`*`] [count` *`count`*`]` command.

NOTE: The only difference in the two commands is the use of the `mep-statistics` and `delay-statistics` keyword.

The fields for these commands are described in Table 16 on page 258.

Table 16: Show Ethernet Delay Command Parameters

Parameter	Parameter Range	Description
`maintenance-domain` *`name`*	Existing MD name	Specifies an existing maintenance domain (MD) to use.
`maintenance-association` *`ma-id`*	Existing MA identifier	Specifies an existing maintenance association (MA) identifier to use.

Table 16: Show Ethernet Delay Command Parameters *(Continued)*

Parameter	Parameter Range	Description
`local-mep` *identifier*	1–8191	When a MEP has been specified, display statistics only for the local MEP.
`remote-mep` *identifier*	1–8191	When a MEP has been specified, display statistics only for the discovered MEP.
`count` *count*	1–100 (default:100)	The number of entries to display in the results table. By default, all 100 entries are displayed if they exist.

NOTE: For each MEP, you will see frame counters for sent and received Ethernet frame delay measurement frames whenever MEP statistics are displayed.

SEE ALSO

Configure a MEP to Generate and Respond to CFM Protocol Messages | 35

Managing ETH-DM Statistics and ETH-DM Frame Counts

IN THIS SECTION

- Displaying ETH-DM Statistics Only | 260
- Displaying ETH-DM Statistics and Frame Counts | 261
- Displaying ETH-DM Frame Counts for MEPs by Enclosing CFM Entity | 262
- Displaying ETH-DM Frame Counts for MEPs by Interface or Domain Level | 263
- Clearing ETH-DM Statistics and Frame Counts | 264

Displaying ETH-DM Statistics Only

IN THIS SECTION

- Purpose | 260
- Action | 260

Purpose

Display ETH-DM statistics.

By default, the `show oam ethernet connectivity-fault-management delay-statistics` command displays ETH-DM statistics for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

Action

- To display the ETH-DM statistics collected for MEPs belonging to MA `ma1` and within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management delay-statistics maintenance-
domain ma1 maintenance-association ma1
```

- To display the ETH-DM statistics collected for ETH-DM sessions for the local MEP `201` belonging to MA `ma2` and within MD `md2`:

```
user@host> show oam ethernet connectivity-fault-management delay-statistics maintenance-
domain md2 maintenance-association ma2 local-mep 201
```

- To display the ETH-DM statistics collected for ETH-DM sessions from local MEPs belonging to MA `ma3` and within MD `md3` to remote MEP `302`:

```
user@host> show oam ethernet connectivity-fault-management delay-statistics maintenance-
domain md3 maintenance-association ma3 remote-mep 302
```

SEE ALSO

show oam ethernet connectivity-fault-management delay-statistics

Displaying ETH-DM Statistics and Frame Counts

IN THIS SECTION

- Purpose | 261
- Action | 261

Purpose

Display ETH-DM statistics and ETH-DM frame counts.

By default, the `show oam ethernet connectivity-fault-management mep-statistics` command displays ETH-DM statistics and frame counts for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

Action

- To display the ETH-DM statistics and ETH-DM frame counts for MEPs in MA `ma1` and within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
md1 maintenance-association ma1
```

- To display the ETH-DM statistics and ETH-DM frame counts for the local MEP `201` in MA `ma2` and within MD `md2`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
md2 maintenance-association ma2 local-mep 201
```

- To display the ETH-DM statistics and ETH-DM frame counts for the local MEP in MD `md3` and within MA `ma3` that participates in an ETH-DM session with the remote MEP `302`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
ma3 maintenance-association ma3 remote-mep 302
```

SEE ALSO

show oam ethernet connectivity-fault-management mep-statistics

Displaying ETH-DM Frame Counts for MEPs by Enclosing CFM Entity

IN THIS SECTION

- Purpose | 262
- Action | 263

Purpose

Display ETH-DM frame counts for CFM maintenance association end points (MEPs).

By default, the `show oam ethernet connectivity-fault-management mep-database` command displays CFM database information for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

NOTE: At the router attached to the initiator MEP for a one-way session, or at the router attached to the receiver MEP for a two-way session, you can only display ETH-DM frame counts.

Action

- To display CFM database information (including ETH-DM frame counts) for all MEPs in MA `ma1` within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
ma1 maintenance-association ma1
```

- To display CFM database information (including ETH-DM frame counts) only for local MEP `201` in MA `ma1` within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
md2 maintenance-association ma2 local-mep 201
```

- To display CFM database information (including ETH-DM frame counts) only for remote MEP `302` in MD `md3` within MA `ma3`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
ma3 maintenance-association ma3 remote-mep 302
```

SEE ALSO

show oam ethernet connectivity-fault-management mep-database

Displaying ETH-DM Frame Counts for MEPs by Interface or Domain Level

IN THIS SECTION

Purpose | 263
Action | 264

Purpose

Display ETH-DM frame counts for CFM maintenance association end points (MEPs).

By default, the `show oam ethernet connectivity-fault-management interfaces` command displays CFM database information for MEPs attached to CFM-enabled Ethernet interfaces on the router or at a maintenance domain level. For Ethernet interfaces that support ETH-DM, any frame counts are also displayed when you specify the `detail` or `extensive` command option.

NOTE: At the router attached to the initiator MEP for a one-way session, or at the router attached to the receiver MEP for a two-way session, you can only display ETH-DM frame counts.

Action

- To display CFM database information (including ETH-DM frame counts) for all MEPs attached to CFM-enabled Ethernet interfaces on the router:

```
user@host> show oam ethernet connectivity-fault-management interfaces detail
```

- To display CFM database information (including ETH-DM frame counts) only for the MEPs attached to CFM-enabled router interface `ge-5/2/9.0`:

```
user@host> show oam ethernet connectivity-fault-management interfaces ge-5/2/9.0 detail
```

- To display CFM database information (including ETH-DM frame counts) only for MEPs enclosed within CFM maintenance domains (MDs) at level `6`:

```
user@host> show oam ethernet connectivity-fault-management interfaces level 6 detail
```

SEE ALSO

show oam ethernet connectivity-fault-management interfaces

Clearing ETH-DM Statistics and Frame Counts

IN THIS SECTION

Purpose | 265

Action | 265

Purpose

Clear the ETH-DM statistics and ETH-DM frame counts.

By default, statistics and frame counts are deleted for all MEPs attached to CFM-enabled interfaces on the router. However, you can filter the scope of the command by specifying an interface name.

Action

- To clear the ETH-DM statistics and ETH-DM frame counts for all MEPs attached to CFM-enabled interfaces on the router:

```
user@host> clear oam ethernet connectivity-fault-management statistics
```

- To clear the ETH-DM statistics and ETH-DM frame counts only for MEPs attached to the logical interface `ge-0/5.9.0`:

```
user@host> clear oam ethernet connectivity-fault-management statistics ge-0/5/9.0
```

SEE ALSO

clear oam ethernet connectivity-fault-management statistics

RELATED DOCUMENTATION

clear oam ethernet connectivity-fault-management statistics

show oam ethernet connectivity-fault-management delay-statistics

show oam ethernet connectivity-fault-management interfaces

show oam ethernet connectivity-fault-management mep-statistics

show oam ethernet connectivity-fault-management mep-database

RELATED DOCUMENTATION

Configure an Iterator Profile | 305

Configure Ethernet Synthetic Loss Measurements | 324

Ethernet Alarm Indication | 341

Inline Transmission Mode | 356

Configuring MEP Interfaces to Support Ethernet Frame Delay Measurements

Ethernet frame delay measurement is a useful tool for providing performance statistics or supporting or challenging Service Level Agreements (SLAs). By default, Ethernet frame delay measurement uses software for timestamping and delay calculations. You can optionally use hardware timing to assist in this process and increase the accuracy of the delay measurement results. This assistance is available on the reception path.

Before you can perform Ethernet frame delay measurements on MX Series routers, you must have done the following:

- Configured Ethernet OAM and CFM correctly
- Prepared the measurement between two compatibly configured MX Series routers
- Enabled the distributed periodic packet management daemon (ppmd)
- Avoided trying to perform Ethernet frame delay measurement on aggregated Ethernet or pseudowire interfaces, which are not supported
- Made sure the hardware-assisted timestamping is supported if that feature is configured

At the end of this configuration, you create two MX Series routers that can perform and display Ethernet frame delay measurements on Ethernet interfaces using optional hardware timestamping. By default, Ethernet frame delay measurement uses software for timestamping and delay calculations. You can optionally use hardware timing to assist in this process and increase the accuracy of the delay measurement results. This assistance is available on the reception path.

To configure hardware-assisted timestamping:

1. To enable Ethernet frame delay measurement hardware assistance on the reception path, include the `hardware-assisted-timestamping` statement at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring]` hierarchy level:

```
[edit]
protocols {
    oam {
        ethernet {
            connectivity-fault-management {
                performance-monitoring {
                    hardware-assisted-timestamping; # Enable timestamping in hardware.
                }
            }
        }
    }
```

2. Ethernet frame delay measurement requires that distributed PPMD is enabled. Before you can gather statistics for Ethernet frame delay measurement, you must make sure that PPMD is configured properly. Without distributed PPMD, delay measurement results are not valid.

   To perform Ethernet frame delay measurement, make sure that the following configuration statement is *NOT* present:

```
[edit routing-options]
ppm {
    no-delegate-processing; # This turns distributed PPMD OFF.
}
```

RELATED DOCUMENTATION

- Ethernet Frame Delay Measurements Overview | 205
- Triggering an Ethernet Frame Delay Measurements Session | 239
- Viewing Ethernet Frame Delay Measurements Statistics | 258
- Example: Configuring One-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 244
- Example: Configuring One-Way Ethernet Frame Delay Measurements with Single-Tagged Interfaces | 244
- Configuring ETH-DM with Untagged Interfaces

Configure Ethernet Frame Loss Measurement

IN THIS SECTION

- Configuring Statistical Frame Loss Measurement for VPLS Connections | 268
- Managing ETH-LM Statistics | 269
- Example: Measuring Ethernet Frame Loss for Single-Tagged LMM/LMR PDUs | 272
- Example: Measuring Ethernet Frame Loss for Dual-Tagged LMM/LMR PDUs | 288

Use this topic to understand more about frame loss measurement and how to configure frame loss measurement.

Currently, loss measurement is not available for Multi-LU cards (MPC3E and MPC4E), and there are no command line interface restrictions for configuration.

Configuring Statistical Frame Loss Measurement for VPLS Connections

Using proactive statistical frame loss measurement, you can monitor VPLS connections on MX Series routers . Statistical frame loss measurement allows you to monitor the quality of Ethernet connections for service level agreements (SLAs). Point-to-point and multipoint-to-multipoint connections configured on MX Series routers can be monitored by registering the connection on an iterator and initiating periodic SLA measurement of frame transmissions on the connections.

Iterators periodically transmit SLA measurement packets using ITU-Y.1731 compliant frames. The iterator sends periodic measurement packets for each of the connections registered to it. These measurement cycles are transmitted in such a way as to not overlap, reducing the processing demands placed on the CPU. The measurement packets are exchanged between the source user network interface (UNI) port and the destination UNI port, providing a sequence of timed performance measurements for each UNI pair. The Frame Loss Ratio (FLR) and connection availability can be computed from these measurements using statistics.

The following steps outline how to configure statistical frame loss measurement for VPLS connections:

1. To configure proactive ETH-DM measurement for a VPLS connection, see "Guidelines for Configuring Routers to Support an ETH-DM Session" on page 224.
2. To enable statistical loss measurement for a VPLS connection, configure an iterator for the VPLS connection using the *sla-iterator-profiles* statement at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring]` hierarchy level. For detailed instructions, see "Configuring an Iterator Profile" on page 306.

3. As part of the iterator configuration, include the `statistical-frame-loss` option for the *measurement-type* statement at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-profiles profile-name]` hierarchy level.
4. Once you have enabled the iterator, you can display the statistical frame loss for a VPLS connection by issuing the `show oam ethernet connectivity-fault-management sla-iterator-statistics sla-iterator identifier maintenance-domain name maintenance-association name local-mep identifier remote-mep identifier` command.

SEE ALSO

Configuring an Iterator Profile | 306

Verifying the Configuration of an Iterator Profile | 309

Managing ETH-LM Statistics

IN THIS SECTION

- Displaying ETH-LM Statistics | 269
- Clearing ETH-LM Statistics | 271

Displaying ETH-LM Statistics

IN THIS SECTION

- Purpose | 269
- Action | 270

Purpose

Display the ETH-LM statistics.

By default, the `show oam ethernet connectivity-fault-management loss-statistics maintenance-domain md-name maintenance-association ma-name` command displays ETH-LM statistics for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

The following list consists of the CFM-related operational mode commands that have been enhanced to display ETH-LM statistics:

- The `show oam ethernet connectivity-fault-management interfaces detail` command is enhanced to display ETH-DM and ETH-LM statistics for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

- The `show oam ethernet connectivity-fault-management mep-statistics` command is enhanced to display ETH-DM and ETH-LM statistics and frame counts for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

- The `show oam ethernet connectivity-fault-management mep-database` command is enhanced to display ETH-DM and ETH-LM frame counters for MEPs in the specified CFM maintenance association (MA) within the specified CFM maintenance domain (MD).

Action

- To display the ETH-LM statistics for all MEPs attached to CFM-enabled interfaces on the router:

```
user@host> show oam ethernet connectivity-fault-management loss-statistics
```

- To display the ETH-DM statistics collected for MEPs belonging to MA `ma1` and within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management delay-statistics maintenance-
domain md1 maintenance-association ma1
```

- To display the ETH-DM statistics and ETH-DM frame counts for MEPs in MA `ma1` and within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
md1 maintenance-association ma1
```

- To display CFM database information (including ETH-DM frame counts) for all MEPs in MA `ma1` within MD `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
md1 maintenance-association ma1
```

SEE ALSO

clear oam ethernet connectivity-fault-management loss-statistics

show oam ethernet connectivity-fault-management delay-statistics

show oam ethernet connectivity-fault-management interfaces

show oam ethernet connectivity-fault-management mep-statistics

show oam ethernet connectivity-fault-management mep-database

show oam ethernet connectivity-fault-management loss-statistics

Clearing ETH-LM Statistics

IN THIS SECTION

Purpose | 271

Action | 271

Purpose

Clear the ETH-LM statistics.

By default, statistics are deleted for all MEPs attached to CFM-enabled interfaces on the router. However, you can filter the scope of the command by specifying an interface name.

Action

- To clear the ETH-LM statistics for all MEPs attached to CFM-enabled interfaces on the router:

```
user@host> clear oam ethernet connectivity-fault-management loss-statistics
```

SEE ALSO

clear oam ethernet connectivity-fault-management loss-statistics

RELATED DOCUMENTATION

Managing ETH-DM Statistics and ETH-DM Frame Counts | 259

Example: Measuring Ethernet Frame Loss for Single-Tagged LMM/LMR PDUs

IN THIS SECTION

- Requirements | 272
- Overview and Topology | 272
- Configuration | 273
- Verification | 286

This example illustrates how to configure Ethernet frame loss measurement (ETH-LM) for single-tagged Loss Measurement Message (LMM)/Loss Measurement Reply (LMR) protocol data units (PDUs). By configuring ETH-LM, you can measure the Ethernet frame loss that occur in your network.

Requirements

This example uses the following hardware and software components:

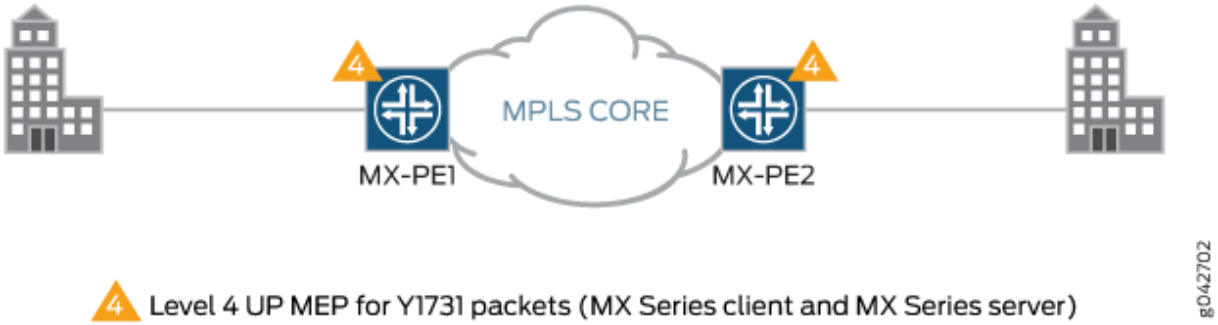
- Two MX Series 5G Universal Routing Platforms with Rev-B Dense Port Concentrators (DPCs)
- Junos OS Release 14.2 or later

Overview and Topology

Junos OS supports Ethernet frame loss measurement (ETH-LM) between maintenance association end points (MEPs) configured on Ethernet physical or logical interfaces on Rev-B Dense Port Concentrators (DPCs) in MX Series routers. Additionally, the Y.1731 functionality supports ETH-LM only for an end-to-end connection that uses Virtual Private Wire Service (VPWS). This example illustrates how to configure ETH-LM for single-tagged LMM/LMR PDUs with input and output VLAN map configured as `swap`.

Figure 19 on page 273 shows the topology used in this example. VPWS service is configured between two MX Series routers, MX-PE1 and MX PE2.

Figure 19: VPWS Service Configured Between Two MX Series Routers



MX-PE1 router has two Ethernet interfaces, `ge-5/0/4` and `ge-5/1/9`. Virtual LAN (VLAN) is configured on `ge-5/0/4` and MPLS is configured on the `ge-5/1/9` interface. The `ge-5/0/4.11` interface is used to configure the Layer 2 virtual circuit with MX-PE2 router. The UP MEP, `mep 2`, is attached to the `ge-5/0/4.11` interface. The three-color policer firewall filter is also configured for the MX-PE1 router.

Similarly, MX-PE2 router has two Ethernet interfaces, `ge-8/0/8` and `ge-8/0/9`. Virtual LAN (VLAN) is configured on `ge-8/0/8` and MPLS is configured on the `ge-8/0/9` interface. The `ge-8/0/8.11` interface is used to configure the Layer 2 virtual circuit with MX-PE1 router. The UP MEP, `mep 1`, is attached to the `ge-8/0/8.11` interface. The three-color policer firewall filter is also configured for the MX-PE2 router.

Configuration

IN THIS SECTION

- CLI Quick Configuration | **273**
- Configuring Router PE1 | **276**
- Configuring Router PE2 | **281**

CLI Quick Configuration

To quickly configure ETH-LM for single-tagged LMM/LMR PDUs, copy the following commands, remove any line breaks, and then paste the commands into the CLI of each device.

On Router PE1:

```
[edit]
set interfaces ge-5/0/4 encapsulation flexible-ethernet-services
set interfaces ge-5/0/4 unit 11 encapsulation vlan-ccc
set interfaces ge-5/0/4 unit 11 layer2-policer input-three-color abc
set interfaces ge-5/0/4 unit 11 family ccc
set interfaces ge-5/1/9 enable
set interfaces ge-5/1/9 unit 0 family inet address 12.1.1.1/24
set interfaces ge-5/1/9 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 4.4.4.4/32
set interfaces ge-5/0/4 flexible-vlan-tagging
set interfaces ge-5/0/4 unit 11 vlan-id 2000
set interfaces ge-5/0/4 unit 11 input-vlan-map swap
set interfaces ge-5/0/4 unit 11 input-vlan-map vlan-id 4094
set interfaces ge-5/0/4 unit 11 output-vlan-map swap
set routing-options router-id 4.4.4.4
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ospf area 0.0.0.0 interface all
set protocols ospf area 0.0.0.0 interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 virtual-circuit-id 1003
set protocols l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 no-control-word
set protocols oam ethernet connectivity-fault-management performance-monitoring delegate-server-
processing
set protocols oam ethernet connectivity-fault-management maintenance-domain md level 4
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 interface ge-5/0/4.11
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 direction up
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 remote-mep 1
set firewall three-color-policer abc logical-interface-policer
set firewall three-color-policer abc two-rate color-blind
set firewall three-color-policer abc two-rate committed-information-rate 10m
set firewall three-color-policer abc two-rate committed-burst-size 1500
```

```
set firewall three-color-policer abc two-rate peak-information-rate 20m
set firewall three-color-policer abc two-rate peak-burst-size 15k
```

On Router PE2:

```
[edit]
set interfaces ge-8/0/8 encapsulation flexible-ethernet-services
set interfaces ge-8/0/8 unit 11 encapsulation vlan-ccc
set interfaces ge-8/0/8 unit 11 layer2-policer input-three-color abc
set interfaces ge-8/0/8 unit 11 family ccc
set interfaces ge-8/0/9 enable
set interfaces ge-8/0/9 unit 0 family inet address 12.1.1.1/24
set interfaces ge-8/0/9 unit 0 family mpls
set interfaces ae0 unit 0 family inet
set interfaces lo0 unit 0 family inet address 3.3.3.3/32
set interfaces ge-8/0/8 flexible-vlan-tagging
set interfaces ge-8/0/8 unit 11 vlan-id 2000
set interfaces ge-8/0/8 unit 11 input-vlan-map swap
set interfaces ge-8/0/8 unit 11 input-vlan-map vlan-id 4094
set interfaces ge-8/0/8 unit 11 output-vlan-map swap
set routing-options router-id 3.3.3.3
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ospf area 0.0.0.0 interface all
set protocols ospf area 0.0.0.0 interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols l2circuit neighbor 4.4.4.4 interface ge-8/0/8.11 virtual-circuit-id 1003
set protocols l2circuit neighbor 3.3.3.3 interface ge-8/0/8.11 no-control-word
set protocols oam ethernet connectivity-fault-management maintenance-domain md level 4
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 interface ge-8/0/8.11
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 direction up
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 remote-mep 2
set firewall three-color-policer abc logical-interface-policer
set firewall three-color-policer abc two-rate color-blind
set firewall three-color-policer abc two-rate committed-information-rate 10m
set firewall three-color-policer abc two-rate committed-burst-size 1500
```

```
set firewall three-color-policer abc two-rate peak-information-rate 20m
set firewall three-color-policer abc two-rate peak-burst-size 15k
```

Configuring Router PE1

Step-by-Step Procedure

To configure Router PE1:

1. Configure the interfaces.

```
[edit]
user@PE1# edit interfaces
[edit interfaces]
user@PE1# set ge-5/0/4 encapsulation flexible-ethernet-services
user@PE1# set ge-5/0/4 unit 11 encapsulation vlan-ccc
user@PE1# set ge-5/0/4 unit 11 layer2-policer input-three-color abc
user@PE1# set ge-5/0/4 unit 11 family ccc
user@PE1# set ge-5/1/9 enable
user@PE1# set ge-5/1/9 unit 0 family inet address 12.1.1.1/24
user@PE1# set ge-5/1/9 unit 0 family mpls
user@PE1# set lo0 unit 0 family inet address 4.4.4.4/32
```

2. Configure the VLAN.

```
[edit interfaces]
user@PE1# set ge-5/0/4 flexible-vlan-tagging
user@PE1# set ge-5/0/4 unit 11 vlan-id 2000
user@PE1# set ge-5/0/4 unit 11 input-vlan-map swap
user@PE1# set ge-5/0/4 unit 11 input-vlan-map vlan-id 4094
user@PE1# set ge-5/0/4 unit 11 output-vlan-map swap
```

3. Configure the router identifier to identify the routing device.

```
[edit]
user@PE1# edit routing-options
[edit routing-options]
user@PE1# set router-id 4.4.4.4
```

4. Configure MPLS, OSPF, and LDP protocols.

```
[edit]
user@PE1# edit protocols
[edit protocols]
user@PE1# set mpls interface all
user@PE1# set mpls interface fxp0.0 disable
user@PE1# set ospf area 0.0.0.0 interface all
user@PE1# set ospf area 0.0.0.0 interface fxp0.0 disable
user@PE1# set ldp interface all
user@PE1# set ldp interface fxp0.0 disable
```

5. Configure the Layer 2 circuit.

```
[edit protocols]
user@PE1# set l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 virtual-circuit-id 1003
user@PE1# set l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 no-control-word
```

6. Configure the MEP.

```
[edit protocols]
user@PE1# set oam ethernet connectivity-fault-management performance-monitoring delegate-
server-processing
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md level 4
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 interface ge-5/0/4.11
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 direction up
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 remote-mep 1
```

7. Configure the firewall.

```
[edit]
user@PE1# edit firewall
[edit firewall]
user@PE1# set three-color-policer abc logical-interface-policer
```

```
user@PE1# set three-color-policer abc two-rate color-blind
user@PE1# set three-color-policer abc two-rate committed-information-rate 10m
user@PE1# set three-color-policer abc two-rate committed-burst-size 1500
user@PE1# set three-color-policer abc two-rate peak-information-rate 20m
user@PE1# set three-color-policer abc two-rate peak-burst-size 15k
```

8. Commit the configuration.

```
[edit]
user@PE1# commit
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show routing-options`, and `show firewall` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@PE1# show interfaces
interfaces {
    ge-5/0/4 {
        flexible-vlan-tagging;
        encapsulation flexible-ethernet-services;
        unit 11 {
            encapsulation vlan-ccc;
            vlan-id 2000;
            input-vlan-map {
                swap;
                vlan-id 4094;
            }
            output-vlan-map swap;
            layer2-policer {
                input-three-color abc;
            }
            family ccc;
        }
    }
    ge-5/1/9 {
        enable;
        unit 0 {
            family inet {
```

```
                address 12.1.1.1/24;
            }
            family mpls;
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 4.4.4.4/32;
            }
        }
    }
}
```

```
user@PE1# show protocols
protocols {
    mpls {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    ospf {
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
    ldp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    l2circuit {
        neighbor 3.3.3.3 {
            interface ge-5/0/4.11 {
                virtual-circuit-id 1003;
                no-control-word;
```

```
            }
        }
    }
    oam {
        ethernet {
            connectivity-fault-management {
                performance-monitoring {
                    delegate-server-processing;
                }
                maintenance-domain md {
                    level 4;
                    maintenance-association ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 2 {
                            interface ge-5/0/4.11;
                            direction up;
                            remote-mep 1;
                        }
                    }
                }
            }
        }
    }
}
```

```
user@PE1# show routing-options
routing-options {
    router-id 4.4.4.4;
}
```

```
user@PE1# show firewall
firewall {
    three-color-policer abc {
        logical-interface-policer;
        two-rate {
            color-blind;
            committed-information-rate 10m;
            committed-burst-size 1500;
```

```
            peak-information-rate 20m;
            peak-burst-size 15k;
        }
    }
}
```

Configuring Router PE2

Step-by-Step Procedure

To configure Router PE2:

1. Configure the interfaces.

```
[edit]
user@PE2# edit interfaces
[edit interfaces]
user@PE2# set ge-8/0/8 encapsulation flexible-ethernet-services
user@PE2# set ge-8/0/8 unit 11 encapsulation vlan-ccc
user@PE2# set ge-8/0/8 unit 11 layer2-policer input-three-color abc
user@PE2# set ge-8/0/8 unit 11 family ccc
user@PE2# set ge-8/0/9 enable
user@PE2# set ge-8/0/9 unit 0 family inet address 12.1.1.1/24
user@PE2# set ge-8/0/9 unit 0 family mpls
user@PE2# set ae0 unit 0 family inet
user@PE2# set lo0 unit 0 family inet address 3.3.3.3/32
```

2. Configure the VLAN.

```
[edit interfaces]
user@PE2# set ge-8/0/8 flexible-vlan-tagging
user@PE2# set ge-8/0/8 unit 11 vlan-id 2000
user@PE2# set ge-8/0/8 unit 11 input-vlan-map swap
user@PE2# set ge-8/0/8 unit 11 input-vlan-map vlan-id 4094
user@PE2# set ge-8/0/8 unit 11 output-vlan-map swap
```

3. Configure the router identifier to identify the routing device.

```
[edit]
user@PE2# edit routing-options
[edit routing-options]
user@PE2# set router-id 3.3.3.3
```

4. Configure MPLS, OSPF, and LDP protocols.

```
[edit]
user@PE2# edit protocols
[edit protocols]
user@PE2# set mpls interface all
user@PE2# set mpls interface fxp0.0 disable
user@PE2# set ospf area 0.0.0.0 interface all
user@PE2# set ospf area 0.0.0.0 interface fxp0.0 disable
user@PE2# set ldp interface all
user@PE2# set ldp interface fxp0.0 disable
```

5. Configure the Layer 2 circuit.

```
[edit protocols]
user@PE2# set l2circuit neighbor 4.4.4.4 interface ge-8/0/8.11 virtual-circuit-id 1003
user@PE2# set l2circuit neighbor 3.3.3.3 interface ge-8/0/8.11 no-control-word
```

6. Configure the MEP.

```
[edit protocols]
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md level 4
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 interface ge-8/0/8.11
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 direction up
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 remote-mep 2
```

7. Configure the firewall.

```
[edit]
user@PE2# edit firewall
[edit firewall]
user@PE2# set three-color-policer abc logical-interface-policer
user@PE2# set three-color-policer abc two-rate color-blind
user@PE2# set three-color-policer abc two-rate committed-information-rate 10m
user@PE2# set three-color-policer abc two-rate committed-burst-size 1500
user@PE2# set three-color-policer abc two-rate peak-information-rate 20m
user@PE2# set three-color-policer abc two-rate peak-burst-size 15k
```

8. Commit the configuration.

```
[edit]
user@PE2# commit
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show routing-options`, and `show firewall` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@PE2# show interfaces
interfaces {
    ge-8/0/8 {
        flexible-vlan-tagging;
        encapsulation flexible-ethernet-services;
        unit 11 {
            encapsulation vlan-ccc;
            vlan-id 2000;
            input-vlan-map {
                swap;
                vlan-id 4094;
            }
            output-vlan-map swap;
            layer2-policer {
                input-three-color abc;
            }
```

```
            family ccc;
        }
    }
    ge-8/0/9 {
        unit 0 {
            family inet {
                address 12.1.1.2/24;
            }
            family mpls;
        }
    }
    ae0 {
        unit 0 {
            family inet;
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 3.3.3.3/32;
            }
        }
    }
}
```

```
user@PE2# show protocols
protocols {
    mpls {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    ospf {
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
```

```
    ldp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    l2circuit {
        neighbor 4.4.4.4 {
            interface ge-8/0/8.11 {
                virtual-circuit-id 1003;
                no-control-word;
            }
        }
    }
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain md {
                    level 4;
                    maintenance-association ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 1 {
                            interface ge-8/0/8.11;
                            direction up;
                            remote-mep 2;
                        }
                    }
                }
            }
        }
    }
}
```

```
user@PE2# show routing-options
routing-options {
```

```
    router-id 3.3.3.3;
}
```

```
user@PE2# show firewall
firewall {
    three-color-policer abc {
        logical-interface-policer;
        two-rate {
            color-blind;
            committed-information-rate 10m;
            committed-burst-size 1500;
            peak-information-rate 20m;
            peak-burst-size 15k;
        }
    }
}
```

Verification

IN THIS SECTION

- Viewing ETH-LM | **286**

To start monitoring the Ethernet frame loss, issue the `monitor ethernet loss-measurement maintenance-domain md maintenance-association ma mep 1` command. Frame loss is calculated by collecting the counter values applicable for ingress and egress service frames where the counters maintain a count of transmitted and received data frames between a pair of MEPs. The loss measurement statistics are retrieved as the output of the `monitor ethernet loss-measurement` command. You can also issue the `show oam ethernet connectivity-fault-management interfaces detail ge-5/0/4.11` command to display ETH-LM statistics.

Viewing ETH-LM

Purpose

View the ETH-LM statistics.

Action

From operational mode, enter the `show oam ethernet connectivity-fault-management interfaces detail ge-5/0/4.11` command.

```
user@PE1> show oam ethernet connectivity-fault-management interfaces detail ge-5/0/4.11
Interface name: ge-5/0/4.11 , Interface status: Active, Link status: Up
  Maintenance domain name: md, Format: string, Level: 4
    Maintenance association name: ma, Format: string
    Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
    Interface status TLV: none, Port status TLV: none
    Connection Protection TLV: no
    MEP identifier: 2, Direction: up, MAC address: 00:24:dc:9b:96:76
    MEP status: running
    Defects:
      Remote MEP not receiving CCM                 : no
      Erroneous CCM received                       : no
      Cross-connect CCM received                   : no
      RDI sent by some MEP                         : no
      Some remote MEP's MAC in error state         : no
    Statistics:
      CCMs sent                                    : 36
      CCMs received out of sequence                : 0
      LBMs sent                                    : 0
      Valid in-order LBRs received                 : 0
      Valid out-of-order LBRs received             : 0
      LBRs received with corrupted data            : 0
      LBRs sent                                    : 0
      LTMs sent                                    : 0
      LTMs received                                : 0
      LTRs sent                                    : 0
      LTRs received                                : 0
      Sequence number of next LTM request          : 0
      1DMs sent                                    : 0
      Valid 1DMs received                          : 0
      Invalid 1DMs received                        : 0
      Out of sync 1DMs received                    : 0
      DMMs sent                                    : 0
      Valid DMMs received                          : 0
      Invalid DMMs received                        : 0
      DMRs sent                                    : 0
      Valid DMRs received                          : 0
```

```
    Invalid DMRs received                        : 0
    LMMs sent                                    : 10
    Valid LMMs received                          : 0
    Invalid LMMs received                        : 0
    LMRs sent                                    : 0
    Valid LMRs received                          : 10
    Invalid LMRs received                        : 0
    SLMs sent                                    : 0
    Valid SLMs received                          : 0
    Invalid SLMs received                        : 0
    SLRs sent                                    : 0
    Valid SLRs received                          : 0
    Invalid SLRs received                        : 0
  Remote MEP count: 1
    Identifier    MAC address        State    Interface
        1     00:05:85:76:e5:30       ok    ge-5/0/4.11
```

Meaning

The Ethernet interface details and statistics are displayed. This output indicates that the `ge-5/0/4.11` interface is active and its link status is `up`. Its maintenance domain name is `md` and its level is `4`. The MEP identifier of the `ge-5/0/4.11` interface is indicated as `2` and its direction is `up`. Under the statistics section, the output indicates that `10` LMMs were sent and `10` valid LMRs were received by the interface.

SEE ALSO

Configure Ethernet Synthetic Loss Measurements | 324

Introduction to OAM Connectivity Fault Management (CFM) | 19

Example: Measuring Ethernet Frame Loss for Dual-Tagged LMM/LMR PDUs

IN THIS SECTION

- Requirements | 289
- Overview and Topology | 289
- Configuration | 290
- Verification | 303

This example illustrates how to configure Ethernet frame loss measurement (ETH-LM) for dual-tagged Loss Measurement Message (LMM)/Loss Measurement Reply (LMR) protocol data units (PDUs). By configuring ETH-LM, you can measure the Ethernet frame loss that occur in your network.

Requirements

This example uses the following hardware and software components:

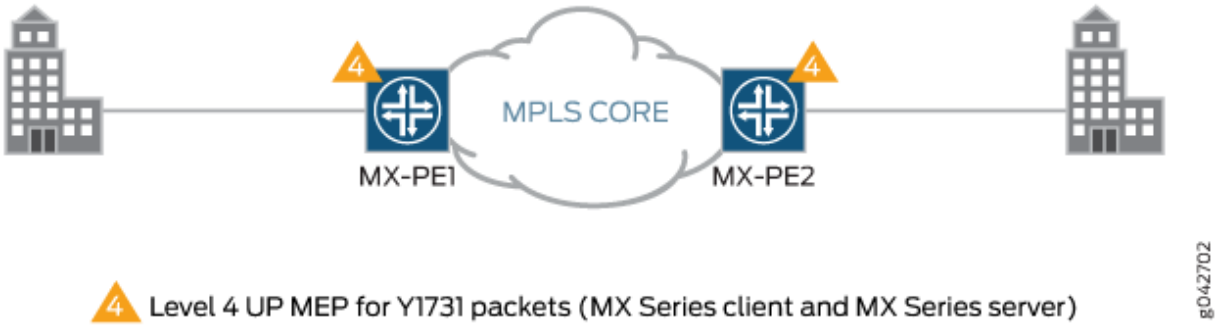
- Two MX Series 5G Universal Routing Platforms with Rev-B Dense Port Concentrators (DPCs)
- Junos OS Release 14.2 or later

Overview and Topology

Junos OS supports Ethernet frame loss measurement (ETH-LM) between maintenance association end points (MEPs) configured on Ethernet physical or logical interfaces on Rev-B Dense Port Concentrators (DPCs) in MX Series routers. Additionally, the Y.1731 functionality supports ETH-LM only for an end-to-end connection that uses Virtual Private Wire Service (VPWS). This example illustrates how to configure ETH-LM for dual tagged LMM/LMR PDUs with input and output VLAN map configured as `swap-swap`.

Figure 20 on page 289 shows the topology used in this example. VPWS service is configured between two MX Sereies routers, MX-PE1 and MX PE2.

Figure 20: VPWS Service Configured Between Two MX Series Routers



MX-PE1 router has two Ethernet interfaces, `ge-5/0/4` and `ge-5/1/9`. Virtual LAN (VLAN) is configured on `ge-5/0/4` and MPLS is configured on the `ge-5/1/9` interface. The `ge-5/0/4.11` interface is used to configure the Layer 2 virtual circuit with MX-PE2 router. The UP MEP, `mep 2`, is attached to the `ge-5/0/4.11` interface. The three-color policer firewall filter is also configured for the MX-PE1 router.

Similarly, MX-PE2 router has two Ethernet interfaces, `ge-8/0/8` and `ge-8/0/9`. Virtual LAN (VLAN) is configured on `ge-8/0/8` and MPLS is configured on the `ge-8/0/9` interface. The `ge-8/0/8.11` interface is used

to configure the Layer 2 virtual circuit with MX-PE1 router. The UP MEP, `mep 1`, is attached to the `ge-8/0/8.11` interface. The three-color policer firewall filter is also configured for the MX-PE2 router.

Configuration

IN THIS SECTION

- CLI Quick Configuration | 290
- Configuring Router PE1 | 292
- Configuring Router PE2 | 298

CLI Quick Configuration

To quickly configure ETH-LM for dual tagged LMM/LMR PDUs, copy the following commands, remove any line breaks, and then paste the commands into the CLI of each device.

On Router PE1:

```
[edit]
set interfaces ge-5/0/4 encapsulation flexible-ethernet-services
set interfaces ge-5/0/4 unit 11 encapsulation vlan-ccc
set interfaces ge-5/0/4 unit 11 layer2-policer input-three-color abc
set interfaces ge-5/0/4 unit 11 family ccc
set interfaces ge-5/1/9 enable
set interfaces ge-5/1/9 unit 0 family inet address 12.1.1.1/24
set interfaces ge-5/1/9 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 4.4.4.4/32
set interfaces ge-5/0/4 flexible-vlan-tagging
set interfaces ge-5/0/4 unit 11 vlan-tags outer 2000 inner 1000
set interfaces ge-5/0/4 unit 11 input-vlan-map swap-swap
set interfaces ge-5/0/4 unit 11 input-vlan-map vlan-id 4094
set interfaces ge-5/0/4 unit 11 input-vlan-map inner-vlan-id 4093
set interfaces ge-5/0/4 unit 11 output-vlan-map swap-swap
set routing-options router-id 4.4.4.4
set protocols mpls interface all
set protocols mpls interface fxp0.0 disable
set protocols ospf area 0.0.0.0 interface all
set protocols ospf area 0.0.0.0 interface fxp0.0 disable
```

```
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 virtual-circuit-id 1003
set protocols l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 no-control-word
set protocols oam ethernet connectivity-fault-management performance-monitoring delegate-server-
processing
set protocols oam ethernet connectivity-fault-management maintenance-domain md level 4
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 interface ge-5/0/4.11
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 direction up
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 2 remote-mep 1
set firewall three-color-policer abc logical-interface-policer
set firewall three-color-policer abc two-rate color-blind
set firewall three-color-policer abc two-rate committed-information-rate 10m
set firewall three-color-policer abc two-rate committed-burst-size 1500
set firewall three-color-policer abc two-rate peak-information-rate 20m
set firewall three-color-policer abc two-rate peak-burst-size 15k
```

On Router PE2:

```
[edit]
set interfaces ge-8/0/8 encapsulation flexible-ethernet-services
set interfaces ge-8/0/8 unit 11 encapsulation vlan-ccc
set interfaces ge-8/0/8 unit 11 layer2-policer input-three-color abc
set interfaces ge-8/0/8 unit 11 family ccc
set interfaces ge-8/0/9 enable
set interfaces ge-8/0/9 unit 0 family inet address 12.1.1.1/24
set interfaces ge-8/0/9 unit 0 family mpls
set interfaces ae0 unit 0 family inet
set interfaces lo0 unit 0 family inet address 3.3.3.3/32
set interfaces ge-8/0/8 flexible-vlan-tagging
set interfaces ge-8/0/8 unit 11 vlan-tags outer 2000 inner 1000
set interfaces ge-8/0/8 unit 11 input-vlan-map swap-swap
set interfaces ge-8/0/8 unit 11 input-vlan-map vlan-id 4094
set interfaces ge-8/0/8 unit 11 input-vlan-map inner-vlan-id 4093
set interfaces ge-8/0/8 unit 11 output-vlan-map swap-swap
set routing-options router-id 3.3.3.3
set protocols mpls interface all
```

```
set protocols mpls interface fxp0.0 disable
set protocols ospf area 0.0.0.0 interface all
set protocols ospf area 0.0.0.0 interface fxp0.0 disable
set protocols ldp interface all
set protocols ldp interface fxp0.0 disable
set protocols l2circuit neighbor 4.4.4.4 interface ge-8/0/8.11 virtual-circuit-id 1003
set protocols l2circuit neighbor 3.3.3.3 interface ge-8/0/8.11 no-control-word
set protocols oam ethernet connectivity-fault-management maintenance-domain md level 4
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 interface ge-8/0/8.11
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 direction up
set protocols oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 remote-mep 2
set firewall three-color-policer abc logical-interface-policer
set firewall three-color-policer abc two-rate color-blind
set firewall three-color-policer abc two-rate committed-information-rate 10m
set firewall three-color-policer abc two-rate committed-burst-size 1500
set firewall three-color-policer abc two-rate peak-information-rate 20m
set firewall three-color-policer abc two-rate peak-burst-size 15k
```

Configuring Router PE1

Step-by-Step Procedure

To configure Router PE1:

1. Configure the interfaces.

```
[edit]
user@PE1# edit interfaces
[edit interfaces]
user@PE1# set ge-5/0/4 encapsulation flexible-ethernet-services
user@PE1# set ge-5/0/4 unit 11 encapsulation vlan-ccc
user@PE1# set ge-5/0/4 unit 11 layer2-policer input-three-color abc
user@PE1# set ge-5/0/4 unit 11 family ccc
user@PE1# set ge-5/1/9 enable
user@PE1# set ge-5/1/9 unit 0 family inet address 12.1.1.1/24
```

```
user@PE1# set ge-5/1/9 unit 0 family mpls
user@PE1# set lo0 unit 0 family inet address 4.4.4.4/32
```

2. Configure the VLAN.

```
[edit interfaces]
user@PE1# set ge-5/0/4 flexible-vlan-tagging
user@PE1# set ge-5/0/4 unit 11 vlan-tags outer 2000 inner 1000
user@PE1# set ge-5/0/4 unit 11 input-vlan-map swap-swap
user@PE1# set ge-5/0/4 unit 11 input-vlan-map vlan-id 4094
user@PE1# set ge-5/0/4 unit 11 input-vlan-map inner-vlan-id 4093
user@PE1# set ge-5/0/4 unit 11 output-vlan-map swap-swap
```

3. Configure the router identifier to identify the routing device.

```
[edit]
user@PE1# edit routing-options
[edit routing-options]
user@PE1# set router-id 4.4.4.4
```

4. Configure MPLS, OSPF, and LDP protocols.

```
[edit]
user@PE1# edit protocols
[edit protocols]
user@PE1# set mpls interface all
user@PE1# set mpls interface fxp0.0 disable
user@PE1# set ospf area 0.0.0.0 interface all
user@PE1# set ospf area 0.0.0.0 interface fxp0.0 disable
user@PE1# set ldp interface all
user@PE1# set ldp interface fxp0.0 disable
```

5. Configure the Layer 2 circuit.

```
[edit protocols]
user@PE1# set l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 virtual-circuit-id 1003
user@PE1# set l2circuit neighbor 3.3.3.3 interface ge-5/0/4.11 no-control-word
```

6. Configure the MEP.

```
[edit protocols]
user@PE1# set oam ethernet connectivity-fault-management performance-monitoring delegate-server-processing
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md level 4
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-association ma continuity-check interval 1s
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-association ma mep 2 interface ge-5/0/4.11
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-association ma mep 2 direction up
user@PE1# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-association ma mep 2 remote-mep 1
```

7. Configure the firewall.

```
[edit]
user@PE1# edit firewall
[edit firewall]
user@PE1# set three-color-policer abc logical-interface-policer
user@PE1# set three-color-policer abc two-rate color-blind
user@PE1# set three-color-policer abc two-rate committed-information-rate 10m
user@PE1# set three-color-policer abc two-rate committed-burst-size 1500
user@PE1# set three-color-policer abc two-rate peak-information-rate 20m
user@PE1# set three-color-policer abc two-rate peak-burst-size 15k
```

8. Commit the configuration.

```
[edit]
user@PE1# commit
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show routing-options`, and `show firewall` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@PE1# show interfaces
interfaces {
    ge-5/0/4 {
        flexible-vlan-tagging;
        encapsulation flexible-ethernet-services;
        unit 11 {
            encapsulation vlan-ccc;
            vlan-tags outer 2000 inner 1000;
            input-vlan-map {
                swap-swap;
                vlan-id 4094;
                inner-vlan-id 4093;
            }
            output-vlan-map swap-swap;
            layer2-policer {
                input-three-color abc;
            }
            family ccc;
        }
    }
    ge-5/1/9 {
        enable;
        unit 0 {
            family inet {
                address 12.1.1.1/24;
            }
            family mpls;
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 4.4.4.4/32;
            }
        }
```

```
    }
}
```

```
user@PE1# show protocols
protocols {
    mpls {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    ospf {
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
    ldp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    l2circuit {
        neighbor 3.3.3.3 {
            interface ge-5/0/4.11 {
                virtual-circuit-id 1003;
                no-control-word;
            }
        }
    }
    oam {
        ethernet {
            connectivity-fault-management {
                performance-monitoring {
                    delegate-server-processing;
                }
                maintenance-domain md {
                    level 4;
```

```
                    maintenance-association ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 2 {
                            interface ge-5/0/4.11;
                            direction up;
                            remote-mep 1;
                        }
                    }
                }
            }
        }
    }
}
```

```
user@PE1# show routing-options
routing-options {
    router-id 4.4.4.4;
}
```

```
user@PE1# show firewall
firewall {
    three-color-policer abc {
        logical-interface-policer;
        two-rate {
            color-blind;
            committed-information-rate 10m;
            committed-burst-size 1500;
            peak-information-rate 20m;
            peak-burst-size 15k;
        }
    }
}
```

Configuring Router PE2

Step-by-Step Procedure

To configure Router PE2:

1. Configure the interfaces.

```
[edit]
user@PE2# edit interfaces
[edit interfaces]
user@PE2# set ge-8/0/8 encapsulation flexible-ethernet-services
user@PE2# set ge-8/0/8 unit 11 encapsulation vlan-ccc
user@PE2# set ge-8/0/8 unit 11 layer2-policer input-three-color abc
user@PE2# set ge-8/0/8 unit 11 family ccc
user@PE2# set ge-8/0/9 enable
user@PE2# set ge-8/0/9 unit 0 family inet address 12.1.1.1/24
user@PE2# set ge-8/0/9 unit 0 family mpls
user@PE2# set ae0 unit 0 family inet
user@PE2# set lo0 unit 0 family inet address 3.3.3.3/32
```

2. Configure the VLAN.

```
[edit interfaces]
user@PE2# set ge-8/0/8 flexible-vlan-tagging
user@PE2# set ge-8/0/8 unit 11 vlan-tags outer 2000 inner 1000
user@PE2# set ge-8/0/8 unit 11 input-vlan-map swap-swap
user@PE2# set ge-8/0/8 unit 11 input-vlan-map vlan-id 4094
user@PE2# set ge-8/0/8 unit 11 input-vlan-map inner-vlan-id 4093
user@PE2# set ge-8/0/8 unit 11 output-vlan-map swap-swap
```

3. Configure the router identifier to identify the routing device.

```
[edit]
user@PE2# edit routing-options
[edit routing-options]
user@PE2# set router-id 3.3.3.3
```

4. Configure MPLS, OSPF, and LDP protocols.

```
[edit]
user@PE2# edit protocols
[edit protocols]
user@PE2# set mpls interface all
user@PE2# set mpls interface fxp0.0 disable
user@PE2# set ospf area 0.0.0.0 interface all
user@PE2# set ospf area 0.0.0.0 interface fxp0.0 disable
user@PE2# set ldp interface all
user@PE2# set ldp interface fxp0.0 disable
```

5. Configure the Layer 2 circuit.

```
[edit protocols]
user@PE2# set l2circuit neighbor 4.4.4.4 interface ge-8/0/8.11 virtual-circuit-id 1003
user@PE2# set l2circuit neighbor 3.3.3.3 interface ge-8/0/8.11 no-control-word
```

6. Configure the MEP.

```
[edit protocols]
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md level 4
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma continuity-check interval 1s
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 interface ge-8/0/8.11
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 direction up
user@PE2# set oam ethernet connectivity-fault-management maintenance-domain md maintenance-
association ma mep 1 remote-mep 2
```

7. Configure the firewall.

```
[edit]
user@PE2# edit firewall
[edit firewall]
user@PE2# set three-color-policer abc logical-interface-policer
user@PE2# set three-color-policer abc two-rate color-blind
user@PE2# set three-color-policer abc two-rate committed-information-rate 10m
```

```
user@PE2# set three-color-policer abc two-rate committed-burst-size 1500
user@PE2# set three-color-policer abc two-rate peak-information-rate 20m
user@PE2# set three-color-policer abc two-rate peak-burst-size 15k
```

8. Commit the configuration.

```
[edit]
user@PE2# commit
```

Results

From configuration mode, confirm your configuration by entering the `show interfaces`, `show protocols`, `show routing-options`, and `show firewall` commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@PE2# show interfaces
interfaces {
    ge-8/0/8 {
        flexible-vlan-tagging;
        encapsulation flexible-ethernet-services;
        unit 11 {
            encapsulation vlan-ccc;
            vlan-tags outer 2000 inner 1000;
            input-vlan-map {
                swap-swap;
                vlan-id 4094;
                inner-vlan-id 4093;
            }
            output-vlan-map swap-swap;
            layer2-policer {
                input-three-color abc;
            }
            family ccc;
        }
    }
    ge-8/0/9 {
        unit 0 {
            family inet {
                address 12.1.1.2/24;
            }
```

```
            family mpls;
        }
    }
    ae0 {
        unit 0 {
            family inet;
        }
    }
    lo0 {
        unit 0 {
            family inet {
                address 3.3.3.3/32;
            }
        }
    }
}
```

```
user@PE2# show protocols
protocols {
    mpls {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    ospf {
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
    ldp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    l2circuit {
        neighbor 4.4.4.4 {
```

```
            interface ge-8/0/8.11 {
                virtual-circuit-id 1003;
                no-control-word;
            }
        }
    }
    oam {
        ethernet {
            connectivity-fault-management {
                maintenance-domain md {
                    level 4;
                    maintenance-association ma {
                        continuity-check {
                            interval 1s;
                        }
                        mep 1 {
                            interface ge-8/0/8.11;
                            direction up;
                            remote-mep 2;
                        }
                    }
                }
            }
        }
    }
}
```

```
user@PE2# show routing-options
routing-options {
    router-id 3.3.3.3;
}
```

```
user@PE2# show firewall
firewall {
    three-color-policer abc {
        logical-interface-policer;
        two-rate {
            color-blind;
            committed-information-rate 10m;
            committed-burst-size 1500;
```

```
            peak-information-rate 20m;
            peak-burst-size 15k;
        }
    }
}
```

Verification

IN THIS SECTION

- Viewing ETH-LM | **303**

To start the Ethernet frame loss measurement session, issue the `monitor ethernet loss-measurement maintenance-domain md maintenance-association ma mep 1` command. Frame loss is calculated by collecting the counter values applicable for ingress and egress service frames where the counters maintain a count of transmitted and received data frames between a pair of MEPs. The loss measurement statistics are retrieved as the output of the `monitor ethernet loss-measurement` command. You can also issue the `show oam ethernet connectivity-fault-management interfaces detail ge-5/0/4.11` command to display ETH-LM statistics.

Viewing ETH-LM

Purpose

View the ETH-LM statistics.

Action

From operational mode, enter the `show oam ethernet connectivity-fault-management interfaces detail ge-5/0/4.11` command.

```
user@PE1> show oam ethernet connectivity-fault-management interfaces detail ge-5/0/4.11
Interface name: ge-5/0/4.11 , Interface status: Active, Link status: Up
  Maintenance domain name: md, Format: string, Level: 4
    Maintenance association name: ma, Format: string
    Continuity-check status: enabled, Interval: 1s, Loss-threshold: 3 frames
    Interface status TLV: none, Port status TLV: none
    Connection Protection TLV: no
```

```
MEP identifier: 2, Direction: up, MAC address: 00:24:dc:9b:96:76
MEP status: running
Defects:
  Remote MEP not receiving CCM                 : no
  Erroneous CCM received                       : no
  Cross-connect CCM received                   : no
  RDI sent by some MEP                         : no
  Some remote MEP's MAC in error state         : no
Statistics:
  CCMs sent                                    : 59
  CCMs received out of sequence                : 0
  LBMs sent                                    : 0
  Valid in-order LBRs received                 : 0
  Valid out-of-order LBRs received             : 0
  LBRs received with corrupted data            : 0
  LBRs sent                                    : 0
  LTMs sent                                    : 0
  LTMs received                                : 0
  LTRs sent                                    : 0
  LTRs received                                : 0
  Sequence number of next LTM request          : 0
  1DMs sent                                    : 0
  Valid 1DMs received                          : 0
  Invalid 1DMs received                        : 0
  Out of sync 1DMs received                    : 0
  DMMs sent                                    : 0
  Valid DMMs received                          : 0
  Invalid DMMs received                        : 0
  DMRs sent                                    : 0
  Valid DMRs received                          : 0
  Invalid DMRs received                        : 0
  LMMs sent                                    : 10
  Valid LMMs received                          : 0
  Invalid LMMs received                        : 0
  LMRs sent                                    : 0
  Valid LMRs received                          : 10
  Invalid LMRs received                        : 0
  SLMs sent                                    : 0
  Valid SLMs received                          : 0
  Invalid SLMs received                        : 0
  SLRs sent                                    : 0
  Valid SLRs received                          : 0
  Invalid SLRs received                        : 0
```

```
Remote MEP count: 1
  Identifier    MAC address        State    Interface
      1     00:05:85:76:e5:30       ok    ge-5/0/4.11
```

Meaning

The Ethernet interface details and statistics are displayed. This output indicates that the `ge-5/0/4.11` interface is active and its link status is `up`. Its maintenance domain name is `md` and its level is `4`. The MEP identifier of the `ge-5/0/4.11` interface is indicated as `2` and its direction is `up`. Under the statistics section, the output indicates that `10` LMMs were sent and `10` valid LMRs were received by the interface.

SEE ALSO

Introduction to OAM Connectivity Fault Management (CFM) | 19

RELATED DOCUMENTATION

ITU-T Y.1731 Ethernet Service OAM Overview | 204

Configure Ethernet Synthetic Loss Measurements | 324

Configure an Iterator Profile

IN THIS SECTION

- Configuring an Iterator Profile | 306
- Verifying the Configuration of an Iterator Profile | 309
- Managing Iterator Statistics | 314
- Configuring a Remote MEP with an Iterator Profile | 322

Use this topic to configure an iterator profile that periodically transmits SLA measurement packets for delay and loss measurement. You can also view and clear the iterator statistics, and configure a remote MEP with an iterator profile.

Configuring an Iterator Profile

You can create an iterator profile with its parameters to periodically transmit SLA measurement packets in the form of ITU-Y.1731-compliant frames for delay measurement or loss measurement.

To create an iterator profile:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management performance-monitoring
```

2. Configure the SLA measurement monitoring iterator:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# edit sla-iterator-profiles
```

3. Configure an iterator profile—for example, i1:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles]
user@host# set i1
```

4. (Optional) Configure the cycle time, which is the amount of time (in milliseconds) between back-to-back transmission of SLA frames for one connection, with values from 10 through 3,600,000. The default value is 1000 ms.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set cycle-time cycle-time-value
```

5. (Optional) Configure the iteration period, which indicates the maximum number of cycles per iteration (the number of connections registered to an iterator cannot exceed this value), with values from 1 through 2000. The default value is 2000.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set iteration-period iteration-period-value
```

6. Configure the measurement type as loss measurement, statistical frame-loss measurement, or two-way delay measurement.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set measurement-type (loss | statistical-frame-loss | two-way-delay)
```

7. (Optional) Configure the calculation weight for delay with values from 1 through 65,535. The default value is 1 (applicable only for two-way delay measurement).

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set calculation-weight delay delay-value
```

8. (Optional) Configure the calculation weight for delay variation with values from 1 through 65,535. The default value is 1 (applicable only for two-way delay measurement).

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set calculation-weight delay-variation delay-variation-value
```

9. (Optional) Configure the threshold value for average frame delay, in microseconds, for two-way Ethernet frame delay measurement (ETH-DM). When the configured threshold for average frame delay is exceeded, an SNMP trap is generated for ETH-DM. The range is from 1 through 4294967295 microseconds.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set avg-fd-twoway-threshold avg-fd-twoway-threshold-value
```

10. (Optional) Configure the threshold value for average frame delay variation, in microseconds, for two-way Ethernet frame delay measurement (ETH-DM). When the configured threshold for

average frame delay variationis exceeded, an SNMP trap is generated for ETH-DM. The range is from 1 through 4294967295 microseconds.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set avg-ifdv-twoway-threshold avg-ifdv-twoway-threshold-value
```

11. (Optional) Configure the threshold value for average frame loss ratio, in milli-percent, in the upward or forward direction for Ethernet loss measurement (ETH-LM) and Ethernet synthetic loss measurement (ETH-SLM). When the configured threshold for average forward frame loss ratio is exceeded, an SNMP trap is generated for ETH-LM and ETH-SLM. The range is from 1 through 100000 milli-percent.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set avg-flr-forward-threshold avg-flr-forward-threshold-value
```

12. (Optional) Configure the threshold value for average frame loss ratio, in milli-percent, in the backward or downstream direction for Ethernet loss measurement (ETH-LM) and Ethernet synthetic loss measurement (ETH-SLM). When the configured threshold for average backward frame loss ratio is exceeded, an SNMP trap is generated for ETH-LM and ETH-SLM. The range is from 1 through 100000 milli-percent.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set avg-flr-backward-threshold avg-flr-backward-threshold-value
```

13. Configure the `disable` statement to stop the iterator (that is, disable the iterator profile).

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set disable
```

14. Verify the configuration.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles]
user@host# show i1

    cycle-time cycle-time-value;
```

```
    iteration-period iteration-period-value;
    measurement-type  (loss | two-way-delay);
    avg-fd-twoway-threshold avg-fd-twoway-threshold-value;
    avg-ifdv-twoway-threshold avg-ifdv-twoway-threshold-value;
    avg-flr-forward-threshold avg-flr-forward-threshold-value;
    avg-flr-backward-threshold avg-flr-backward-threshold-value;

    calculation-weight {
        delay delay-weight;
        delay-variation delay-variation-weight;
    }
    calculation-weight {
        delay delay-weight;
        delay-variation delay-variation-weight;
    }
```

SEE ALSO

Proactive Mode for SLA Measurement | 214

Verifying the Configuration of an Iterator Profile

IN THIS SECTION

- Displaying the Configuration of an Iterator Profile for Two-way Delay Measurement | 310
- Displaying the Configuration of an Iterator Profile for Loss Measurement | 311
- Displaying the Configuration of a Remote MEP with an Iterator Profile | 312
- Disabling an Iterator Profile | 313

The following topics illustrate the configuration of an iterator profile for a two-way delay measurement, for loss measurement, and for a remote maintenance association end point (MEP). The topics also illustrate disabling an iterator profile with the `disable` statement for two-way measurement and deactivating an iterator profile with the `deactivate` command for a remote MEP.

Displaying the Configuration of an Iterator Profile for Two-way Delay Measurement

IN THIS SECTION

- Purpose | 310
- Action | 310
- Meaning | 311

Purpose

Display the configuration of an iterator profile for two-way delay measurement as configured in the "Configuring an Iterator Profile" on page 306 topic with the following values:

- `profile-name`—**i1**
- `cycle-time`—**1000** milliseconds
- `iteration-period`—**2000** cycles per second
- `delay`—**1**
- `delay-variation`—**1**:

Action

To display information about the iterator profile, run the `show` command at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-profiles]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-
profiles]
user@host# show
i1 {
    cycle-time 1000;
    iteration-period 2000;
    measurement-type two-way-delay;
    calculation-weight {
        delay 1;
        delay-variation 1;
```

```
    }
}
```

Meaning

The configuration for an iterator profile for two-way measurement is displayed as expected with set values.

Displaying the Configuration of an Iterator Profile for Loss Measurement

IN THIS SECTION

- Purpose | **311**
- Action | **311**
- Meaning | **312**

Purpose

Display the configuration of an iterator profile for loss measurement as configured in the "Configuring an Iterator Profile" on page 306 topic with the following values:

- `profile-name`—**12**
- `cycle-time`—**1000** milliseconds
- `iteration-period`—**2000** cycles per second

Action

To display information about the iterator profile, run the `show` command at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-profiles]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-
profiles]
user@host# show
12 {
    cycle-time 1000;
    iteration-period 2000;
```

```
    measurement-type loss;

}
```

Meaning

The configuration for an iterator profile for loss measurement is displayed as expected with set values.

Displaying the Configuration of a Remote MEP with an Iterator Profile

IN THIS SECTION

- Purpose | 312
- Action | 313
- Meaning | 313

Purpose

Display the configuration of a remoteMEP as configured in the "Configuring a Remote MEP with an Iterator Profile" on page 322 topic with the following values:

- `profile-name`—**i3**
- `maintenance-domain`—**default-1**
- `maintenance-association`—**1**
- `short-name-format`—**2octet**
- `mep`—**1**
- `remote-mep`—**1**
- `data-tlv-size`—**1**
- `iteration-count`—**1**
- `priority`—**1**

Action

To display information about the remote MEP, run the `show` command at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain default-1 maintenance association ma1 mep 1 remote-mep 1]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain default-1
maintenance association 1  short-name-format 2octet mep 1 remote-mep 1]
user@host# show
sla-iterator-profile i3 {
    data-tlv-size 1;
    iteration-count 1;
    priority 1;
}
```

Meaning

The configuration for a remote MEP for two-way measurement is displayed as expected with set values.

Disabling an Iterator Profile

IN THIS SECTION

- Purpose | 313
- Action | 314

Purpose

To disable an iterator profile for two-way delay measurement and for a remote MEP.

Action

- To disable an iterator profile (for example, i1) with the `disable` configuration command for two-way measurement at the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-iterator-profiles i1]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# disable
```

- To disable an iterator profile for a remote MEP (for example, i2) with the `deactivate` configuration command at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain default-1 maintenance association ma1 mep 1 remote-mep 1]` hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain default-1
maintenance association ma1 mep 1 remote-mep 1]
user@host# deactivate sla-iterator-profile i2
```

RELATED DOCUMENTATION

Proactive Mode for SLA Measurement | 214

Managing Iterator Statistics

IN THIS SECTION

Displaying Iterator Statistics | 314
Clearing Iterator Statistics | 321

Displaying Iterator Statistics

IN THIS SECTION

Purpose | 315

- Action | 315

Purpose

Retrieve and display iterator statistics.

Muliple iterators can be associated with a remote MEP. However, by default, only one result pertaining to one iterator profile is displayed.

Action

- To display the iterator statistics for remote MEP `1` and iterator profile `i1` with MEPs belonging to the maintenance association `ma1` and within the maintenance domain `default-1` (here, the iterator profile `i1` is configured for two-way delay measurement):

```
user@host> show oam ethernet connectivity-fault-management sla-iterator-statistics sla-
iterator i1 maintenance-domain default-1 maintenance-association ma1 local-mep 1 remote-mep 1
Iterator statistics:
Maintenance domain: md6, Level: 6
Maintenance association: ma6, Local MEP id: 1000
Remote MEP id: 103, Remote MAC address: 00:90:69:0a:43:92
Iterator name: i1, Iterator Id: 1
Iterator cycle time: 10ms, Iteration period: 1 cycles
Iterator status: running, Infinite iterations: true
Counter reset time: 2010-03-19 20:42:39 PDT (2d 18:24 ago)
Reset reason: Adjacency flap

Iterator delay measurement statistics:
Delay weight: 1, Delay variation weight: 1
DMM sent                                        : 23898520
DMM skipped for threshold hit                   : 11000
DMM skipped for threshold hit window            : 0
DMR received                                    : 23851165
DMR out of sequence                             : 1142
DMR received with invalid time stamps           : 36540
Average two-way delay                           : 129 usec
Average two-way delay variation                 : 15 usec
Average one-way forward delay variation         : 22 usec
Average one-way backward delay variation        : 22 usec
```

```
Weighted average two-way delay                    : 134 usec
Weighted average two-way delay variation          : 8 usec
Weighted average one-way forward delay variation  : 6 usec
Weighted average one-way backward delay variation : 2 usec
```

Output fields are listed in the approximate order in which they appear.

Table 17: Displaying Iterator Statistics for Ethernet Delay Measurement Output Fields

Output Field Name	Output Field Description
`Maintenance domain`	Maintenance domain name.
`Level`	Maintenance domain level configured.
`Maintenance association`	Maintenance association name.
`Local MEP id`	Numeric identifier of the local MEP.
`Remote MEP id`	Numeric identifier of the remote MEP.
`Remote MAC address`	Unicast MAC address of the remote MEP.
`Iterator name`	Name of iterator.
`Iterator Id`	Numeric identifier of the iterator.
`Iterator cycle time`	Number of cycles (in milliseconds) taken between back-to-back transmission of SLA frames for this connection
`Iteration period`	Maximum number of cycles per iteration
`Iterator status`	Current status of iterator whether running or stopped.
`Infinite iterations`	Status of iteration as infinite or finite.

Table 17: Displaying Iterator Statistics for Ethernet Delay Measurement Output Fields *(Continued)*

Output Field Name	Output Field Description
`Counter reset time`	Date and time when the counter was reset.
`Reset reason`	Reason to reset counter.
`Delay weight`	Calculation weight of delay.
`Delay variation weight`	Calculation weight of delay variation.
`DMM sent`	Delay measurement message (DMM) PDU frames sent to the peer MEP in this session.
`DMM skipped for threshold hit`	Number of DMM frames sent to the peer MEP in this session skipped during threshold hit.
`DMM skipped for threshold hit window`	Number of DMM frames sent to the peer MEP in this session skipped during the last threshold hit window.
`DMR received`	Number of delay measurement reply (DMR) frames received.
`DMR out of sequence`	Total number of DMR out of sequence packets received.
`DMR received with invalid time stamps`	Total number of DMR frames received with invalid timestamps.
`Average two-way delay`	Average two-way frame delay for the statistics displayed.
`Average two-way delay variation`	Average two-way “frame jitter” for the statistics displayed.
`Average one-way forward delay variation`	Average one-way forward delay variation for the statistics displayed in microseconds.

Table 17: Displaying Iterator Statistics for Ethernet Delay Measurement Output Fields *(Continued)*

Output Field Name	Output Field Description
`Average one-way backward delay variation`	Average one-way backward delay variation for the statistics displayed in microseconds.
`Weighted average two-way delay`	Weighted average two-way delay for the statistics displayed in microseconds.
`Weighted average two-way delay variation`	Weighted average two-way delay variation for the statistics displayed in microseconds.
`Weighted average one-way forward delay variation`	Weighted average one-way forward delay variation for the statistics displayed in microseconds.
`Weighted average one-way backward delay variation`	Weighted average one-way backward delay variation for the statistics displayed in microseconds.

- To display the iterator statistics for remote MEP `1` and iterator profile `i2` with MEPs belonging to the maintenance association `ma1` and within the maintenance domain `default-1` (here, the iterator profile `i1` is configured for loss measurement):

```
user@host> show oam ethernet connectivity-fault-management sla-iterator-statistics sla-
iterator i2 maintenance-domain default-1 maintenance-association ma1 local-mep 1 remote-mep 1
Iterator statistics:
Maintenance domain: md6, Level: 6
Maintenance association: ma6, Local MEP id: 1000
Remote MEP id: 103, Remote MAC address: 00:90:69:0a:43:92
Iterator name: i2, Iterator Id: 2
Iterator cycle time: 1000ms, Iteration period: 2000 cycles
Iterator status: running, Infinite iterations: true
Counter reset time: 2010-03-19 20:42:39 PDT (2d 18:25 ago)
Reset reason: Adjacency flap

Iterator loss measurement statistics:
LMM sent                              : 238970
LMM skipped for threshold hit         : 60
LMM skipped for threshold hit window  : 0
```

```
LMR received                          : 238766
LMR out of sequence                   : 43

Accumulated transmit statistics:
Near-end (CIR)                        : 0
Far-end (CIR)                         : 0
Near-end (EIR)                        : 0
Far-end (EIR)                         : 0

Accumulated loss statistics:
Near-end (CIR)                        : 0 (0.00%)
Far-end (CIR)                         : 0 (0.00%)
Near-end (EIR)                        : 0 (0.00%)
Far-end (EIR)                         : 0 (0.00%)

Last loss measurement statistics:
Near-end (CIR)                        : 0
Far-end (CIR)                         : 0
Near-end (EIR)                        : 0
Far-end (EIR)                         : 0
```

Output fields are listed in the approximate order in which they appear.

Table 18: Displaying Iterator Statistics for Ethernet Loss Measurement Output Fields

Output Field Name	Output Field Description
`Maintenance domain`	Maintenance domain name.
`Level`	Maintenance domain level configured.
`Maintenance association`	Maintenance association name.
`Local MEP id`	Numeric identifier of the local MEP.
`RemoteMEP identifier`	Numeric identifier of the remote MEP.
`Remote MAC address`	Unicast MAC address of the remote MEP.

Table 18: Displaying Iterator Statistics for Ethernet Loss Measurement Output Fields *(Continued)*

Output Field Name	Output Field Description
`Iterator name`	Name of iterator.
`Iterator Id`	Numeric identifier of the iterator.
`Iterator cycle time`	Number of cycles (in milliseconds) taken between back-to-back transmission of SLA frames for this connection
`Iteration period`	Maximum number of cycles per iteration
`Iterator status`	Current status of iterator whether running or stopped.
`Infinite iterations`	Status of iteration as infinite or finite.
`Counter reset time`	Date and time when the counter was reset.
`Reset reason`	Reason to reset counter.
`LMM sent`	Number of loss measurement message (LMM) PDU frames sent to the peer MEP in this session.
`LMM skipped for threshold hit`	Number of LMM frames sent to the peer MEP in this session skipped during threshold hit.
`LMM skipped for threshold hit window`	Number of LMM frames sent to the peer MEP in this session skipped during the last threshold hit window.
`LMR received`	Number of LMRs frames received.
`LMR out of sequence`	Total number of LMR out of sequence packets received.
`Near-end (CIR)`	Frame loss associated with ingress data frames for the statistics displayed.

Table 18: Displaying Iterator Statistics for Ethernet Loss Measurement Output Fields *(Continued)*

Output Field Name	Output Field Description
`Far-end (CIR)`	Frame loss associated with egress data frames for the statistics displayed.
`Near-end (EIR)`	Frame loss associated with ingress data frames for the statistics displayed.
`Far-end (EIR)`	Frame loss associated with egress data frames for the statistics displayed.

SEE ALSO

Proactive Mode for SLA Measurement | **214**

show oam ethernet connectivity-fault-management sla-iterator-statistics

clear oam ethernet connectivity-fault-management sla-iterator-statistics

Clearing Iterator Statistics

IN THIS SECTION

- Purpose | **321**
- Action | **322**

Purpose

Clear iterator statistics.

Multiple iterators can be associated with remote MEP. However, by default, only one result pertaining to one iterator profile can be cleared.

Action

- To clear the iterator statistics for remote MEP 1 and iterator profile i1 with MEPs belonging to the maintenance association ma1 and within the maintenance domain default-1:

```
user@host> clear oam ethernet connectivity-fault-management sla-iterator-statistics sla-
iterator i1 maintenance-domain default-1 maintenance-association ma1 local-mep 1 remote-mep 1
```

- To clear the iterator statistics for remote MEP 1 and iterator profile i2 with MEPs belonging to the maintenance association ma1 and within the maintenance domain default-1:

```
user@host> clear oam ethernet connectivity-fault-management sla-iterator-statistics sla-
iterator i2 maintenance-domain default-1 maintenance-association ma1 local-mep 1 remote-mep 1
```

SEE ALSO

> Proactive Mode for SLA Measurement | 214
>
> *show oam ethernet connectivity-fault-management sla-iterator-statistics*
>
> *clear oam ethernet connectivity-fault-management sla-iterator-statistics*

Configuring a Remote MEP with an Iterator Profile

You can associate a remote maintenance association end point (MEP) with more than one iterator profile.

To configure a remote MEP with an iterator profile:

1. In configuration mode, go to the following hierarchy level:

```
user@host# edit protocols oam ethernet connectivity-fault-management maintenance-domain md-
name maintenance-association ma-name mep mep-id
```

2. Configure the remote MEP with values from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id]
user@host# set remote-mep remote-mep-id
```

3. Set the iterator profile.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id]
user@host# set sla-iterator-profile profile-name
```

4. (Optional) Set the size of the data TLV portion of the Y.1731 data frame with values from 1 through 1400 bytes. The default value is 1.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id sla-iterator-profile
profile-name]
user@host# set data-tlv-size size
```

5. (Optional) Set the iteration count, which indicates the number of iterations for which this connection should partake in the iterator for acquiring SLA measurements, with values from 1 through 65,535. The default value is 0 (that is, infinite iterations).

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id sla-iterator-profile
profile-name]
user@host# set iteration-count count-value
```

6. (Optional) Set the priority, which is the `vlan-pcp` value that is sent in the Y.1731 data frames, with values from 0 through 7. The default value is 0.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id sla-iterator-profile
profile-name]
user@host# set priority priority-value
```

7. Verify the configuration.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id]
user@host# show
sla-iterator-profile profile-name  {
    data-tlv-size size;
    iteration-count count-value;
```

```
    priority priority-value;
}
```

SEE ALSO

Proactive Mode for SLA Measurement | **214**

remote-mep

RELATED DOCUMENTATION

ITU-T Y.1731 Ethernet Service OAM Overview | **204**

Configure Ethernet Frame Delay Measurement Sessions | **224**

Configure Ethernet Synthetic Loss Measurements

IN THIS SECTION

- Guidelines for Configuring ETH-SLM | **324**
- Starting a Proactive ETH-SLM Session | **326**
- Starting an On-Demand ETH-SLM Session | **331**
- Managing ETH-SLM Statistics and ETH-SLM Frame Counts | **332**
- Troubleshooting Failures with ETH-SLM | **339**

Use this topic to understand the guidelines for configuring synthetic loss measurement and how to start a synthetic loss measurement session. There are two types of synthetic loss measurement sessions: proactive and On-Demand. This topic describes both. Also, the topic shows you how to view and clear the synthetic loss measurement statistics and how to troubleshoot failures with SLM.

Guidelines for Configuring ETH-SLM

Keep the following points in mind when you configure the ETH-SLM functionality:

- The monitoring application for Ethernet OAM is initiated in the primary Routing Engine. When a stateful switchover process occurs, the monitoring application is disabled. For on-demand ETH-SLM, *graceful Routing Engine switchover* (GRES) support is not applicable. For proactive ETH-SLM, the service-level agreement (SLA) iterators are restored during a stateful switchover process. If the adjacencies do not time out, the ETH-SLM statistics are preserved and proactive ETH-SLM supports GRES.

- ETH-SLM is initiated only when the MEP session is up. Unified in-service software upgrade (ISSU) support for ETH-SLM depends on the unified ISSU support for CFM. For CFM, unified ISSU is supported using the loss threshold TLV to avoid CFM connectivity loss during the upgrade. The receiving or the destination MEP increases the threshold time during the termination of sessions. If you start a unified ISSU operation when on-demand ETH-SLM is in progress, the SLM request and reply messages are lost at the local Packet Forwarding Engine.

  When an on-demand ETH-SLM is requested, if the local source MEP undergoes a unified ISSU, a message is displayed stating that the MEP is undergoing a unified ISSU. If the remote MEP is undergoing a unified ISSU (detected through the loss threshold TLV), a message is displayed stating that the remote MEP is undergoing a unified ISSU. Also, if it is not possible to identify whether unified ISSU is in progress on a remote MEP, the SLM packets are lost at the system where unified ISSU is in progress and the loss calculation results do not provide a valid cause for the loss. Unified ISSU is not supported for both on-demand and proactive ETH-SLM.

- The maximum number of SLA iterator profiles that can be configured in the system is 255.

- ETH-SLM is not supported for virtual private LAN service (VPLS) (point-to-multipoint measurements are not supported). The ETH-SLM frames are not generated with multicast class 1 destination address. Similarly, ETH-SLM does not respond to ETH-SLM requests with multicast DA. ETH-SLM for VPLS for point-to-point Ethernet connection is supported using directed unicast destination MAC addresses, although point-to-multipoint topologies are not supported.

- A unicast destination address may be used in provisioned environments for point-to-point connections. However, it requires that the unicast destination address of the downstream MEP must have been configured on the MEP transmitting an alarm indication signal (AIS).

- ETH-SLM is not supported on downstream MEPs on label-switched interfaces (LSIs).

- ETH-SLM is supported on aggregated Ethernet (ae) interfaces

- The number of ETH-SLM sessions for proactive ETH-SLM that can be supported is limited to the total number of iterators that can be supported in the system. This limitation includes the iterator support for other measurement types such as loss, statistical frame loss, and two-way delay. A new iterator type, SLM, is added to support ETH-SLM. The total number of SLA iterators that you can configure in the system is equal to the total number of iterations supported in the system.

- For on-demand SLM, the minimum period between two SLM requests is 100 milliseconds.

- For proactive SLM, the minimum period between two SLM requests is 10 milliseconds for distributed mode and 100 milliseconds for non-distributed mode.
- ETH-SLM frames are always marked as drop-ineligible in compliance with the ITU-T Y.1731 standard.

SEE ALSO

> Ethernet Synthetic Loss Measurement Overview | **217**
>
> *monitor ethernet synthetic-loss-measurement*

Starting a Proactive ETH-SLM Session

IN THIS SECTION

- Configuring MEP Interfaces | **326**
- Configuring an Iterator Profile for ETH-SLM | **327**
- Associating the Iterator Profile with MEPs for ETH-SLM | **329**

To start a proactive Ethernet synthetic loss measurement (ETH-SLM) session, you must configure the Ethernet interfaces on maintenance association end points (MEPs) on which packets transmitted with synthetic frame loss need to be analyzed. You must then create an iterator profile to transmit service-level agreement (SLA) measurement packets for ETH-SLM and associate the local and remote MEPs with the profile.

Configuring MEP Interfaces

Before you can start an Ethernet synthetic frame loss measurement session across an Ethernet service, you must configure two ACX Series routers to support ETH-SLM.

To configure an Ethernet interface on an ACX Series router to support ETH-SLM:

1. On each router, configure two physical or logical Ethernet interfaces connected by a VLAN. The following configuration is typical for single-tagged logical interfaces:

```
[edit interfaces]
interface {
    ethernet-interface-name {
        vlan-tagging;
```

```
        unit logical-unit-number {
            vlan-id vlan-id; # Both interfaces on this VLAN
        }
    }
}
```

Both interfaces will use the same VLAN ID.

2. On each router, attach peer MEPs to the two interfaces. The following configuration is typical:

```
[edit protocols]
oam {
    ethernet {
        connectivity-fault-management {
            maintenance-domain md-name { # On both routers
                level number;
                maintenance-association ma-name { # On both routers
                    continuity-check {
                        interval 100ms;
                        hold-interval 1;
                    }
                    mep mep-id { # Attach to VLAN interface
                        auto-discovery;
                        direction (up | down);
                        interface interface-name;
                        priority number;
                    }
                }
            }
        }
    }
}
```

Configuring an Iterator Profile for ETH-SLM

You can create an iterator profile with its parameters to periodically transmit SLA measurement packets in the form of ITU-Y.1731-compliant frames for synthetic loss measurement.

NOTE: ACX5048 and ACX5096 routers supports iterator cycle time of only 1 second and above.

To create an iterator profile:

1. In configuration mode, go to the following hierarchy level:

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management performance-monitoring
```

2. Configure the SLA measurement monitoring iterator:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# edit sla-iterator-profiles
```

3. Configure an iterator profile—for example, i1:

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles]
user@host# set i1
```

4. (Optional) Configure the cycle time, which is the amount of time (in milliseconds) between back-to-back transmission of SLA frames for one connection, with a value from 10 through 3,600,000. The default value is 1000 ms.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set cycle-time cycle-time-value
```

5. (Optional) Configure the iteration period, which indicates the maximum number of cycles per iteration (the number of connections registered to an iterator cannot exceed this value), with a value from 1 through 2000. The default value is 2000.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set iteration-period iteration-period-value
```

6. Configure the measurement type as synthetic loss measurement.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set measurement-type slm
```

7. Configure the `disable` statement to stop the iterator (that is, disable the iterator profile).

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles i1]
user@host# set disable
```

8. Verify the configuration.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring sla-
iterator-profiles]
user@host# show i1

    cycle-time cycle-time-value;
    iteration-period iteration-period-value;
    measurement-type slm;
```

Associating the Iterator Profile with MEPs for ETH-SLM

You can associate a remote maintenance association end point (MEP) with more than one iterator profile.

To configure a remote MEP with an iterator profile:

1. In configuration mode, go to the following hierarchy level:

```
user@host# edit protocols oam ethernet connectivity-fault-management maintenance-domain md-
name maintenance-association ma-name mep mep-id
```

2. Configure the remote MEP ID with a value from 1 through 8191.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id]
user@host# set remote-mep remote-mep-id
```

3. Set the iterator profile.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id]
user@host# set sla-iterator-profile profile-name
```

4. (Optional) Set the size of the data TLV portion of the Y.1731 data frame with a value from 1 through 1400 bytes. The default value is 1.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id sla-iterator-profile
profile-name]
user@host# set data-tlv-size size
```

5. (Optional) Set the iteration count, which indicates the number of iterations for which this connection should partake in the iterator for acquiring SLA measurements, with a value from 1 through 65,535. The default value is 0 (that is, infinite iterations).

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id sla-iterator-profile
profile-name]
user@host# set iteration-count count-value
```

6. (Optional) Set the priority, which is the `vlan-pcp` value that is sent in the Y.1731 data frames, with a value from 0 through 7. The default value is 0.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id sla-iterator-profile
profile-name]
user@host# set priority priority-value
```

7. Verify the configuration.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id remote-mep remote-mep-id]
user@host# show
sla-iterator-profile profile-name  {
    data-tlv-size size;
    iteration-count count-value;
```

```
    priority priority-value;
}
```

RELATED DOCUMENTATION

Ethernet Synthetic Loss Measurement Overview | 217

monitor ethernet synthetic-loss-measurement

Starting an On-Demand ETH-SLM Session

To start an on-demand Ethernet synthetic loss measurement (ETH-SLM) session, type the `monitor ethernet synthetic-loss-measurement one-way` command in operational mode, and specify the peer MEP by its MAC address or by its MEP identifier.

For example:

```
user@host> monitor ethernet synthetic-loss-measurement 00:05:85:73:39:4a maintenance-domain md6
maintenance-association ma6 count 10
ETH-SLM request to 00:05:85:73:39:4a, interface ge-1/0/0.0
      Synthetic Loss measurement statistics:
       SLM packets sent                          : 100
       SLR packets received                      : 100
       Accumulated SLM statistics:
       Local TXFC1 value                         : 100
       Local RXFC1 value                         : 100
       Last Received SLR frame TXFCf(tc)         : 100
       Last Received SLR frame TXFCb(tc)         : 100
       SLM Frame Loss:
       Frame Loss (far-end)                      : 0 (0.00 %)
       Frame Loss (near-end)                     : 0 (0.00 %)
```

NOTE: If you attempt to monitor delays to a nonexistent MAC address, you must press **Ctrl** + **C** to explicitly quit the `monitor ethernet synthetic-loss-measurement` command and return to the CLI command prompt.

SEE ALSO

Ethernet Synthetic Loss Measurement Overview | 217

monitor ethernet synthetic-loss-measurement

Managing ETH-SLM Statistics and ETH-SLM Frame Counts

IN THIS SECTION

- Displaying ETH-SLM Statistics Only | 332
- Displaying ETH-SLM Statistics and Frame Counts | 333
- Displaying ETH-SLM Frame Counts for MEPs by Enclosing CFM Entity | 335
- Displaying ETH-SLM Frame Counts for MEPs by Interface or Domain Level | 336
- Clearing ETH-SLM Statistics and Frame Counts | 337
- Clearing Iterator Statistics | 338

Displaying ETH-SLM Statistics Only

IN THIS SECTION

- Purpose | 332
- Action | 333
- Meaning | 333

Purpose

Display on-demand ETH-SLM statistics.

By default, the `show oam ethernet connectivity-fault-management synthetic-loss-statistics` command displays on-demand ETH-SLM statistics for MEPs in the specified CFM maintenance association within the specified CFM maintenance domain.

Action

- To display the on-demand ETH-SLM statistics collected for MEPs belonging to maintenance association `ma1` within maintenance domain `md1`:

```
user@host> show oam ethernet connectivity-fault-management synthetic-loss-statistics
maintenance-domain md1 maintenance-association ma1
```

- To display the on-demand ETH-SLM statistics collected for ETH-SLM sessions for the local MEP `201` belonging to maintenance association `ma2` within maintenance domain `md2`:

```
user@host> show oam ethernet connectivity-fault-management synthetic-loss-statistics
maintenance-domain md2 maintenance-association ma2 local-mep 201
```

- To display the on-demand ETH-SLM statistics collected for ETH-SLM sessions from local MEPs belonging to maintenance association `ma3` within maintenance domain `md3` to the remote MEP `302`:

```
user@host> show oam ethernet connectivity-fault-management synthetic-loss-statistics
maintenance-domain md3 maintenance-association ma3 remote-mep 302
```

Meaning

The output displays on-demand ETH-SLM statistics for MEPs in the specified maintenance association within the specified maintenance domain. For details about the output of this command and the descriptions of the output fields, see `show oam ethernet connectivity-fault-management synthetic-loss-statistics`.

SEE ALSO

show oam ethernet connectivity-fault-management synthetic-loss-statistics

Displaying ETH-SLM Statistics and Frame Counts

IN THIS SECTION

- Purpose | 334
- Action | 334

- Meaning | 334

Purpose

Display on-demand ETH-SLM statistics and ETH-SLM frame counts.

By default, the `show oam ethernet connectivity-fault-management mep-statistics` command displays on-demand ETH-SLM statistics and frame counts for MEPs in the specified CFM maintenance association within the specified CFM maintenance domain.

Action

- To display the on-demand ETH-SLM statistics and ETH-SLM frame counts for MEPs in maintenance association `ma1` within maintenance domain `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
md1 maintenance-association ma1
```

- To display the on-demand ETH-SLM statistics and ETH-SLM frame counts for the local MEP `201` in maintenance association `ma2` within maintenance domain `md2`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
md2 maintenance-association ma2 local-mep 201
```

- To display the on-demand ETH-SLM statistics and ETH-SLM frame counts for the local MEP in maintenance association `ma3` within maintenance domain `md3` that participates in an ETH-SLM session with the remote MEP `302`:

```
user@host> show oam ethernet connectivity-fault-management mep-statistics maintenance-domain
ma3 maintenance-association ma3 remote-mep 302
```

Meaning

The output displays on-demand ETH-SLM statistics and ETH-SLM frame counts for MEPs in the specified maintenance association within the specified maintenance domain. For details about the

output of this command and the descriptions of the output fields, see `show oam ethernet connectivity-fault-management mep-statistics`.

SEE ALSO

show oam ethernet connectivity-fault-management mep-statistics

Displaying ETH-SLM Frame Counts for MEPs by Enclosing CFM Entity

IN THIS SECTION

- Purpose | 335
- Action | 335
- Meaning | 336

Purpose

Display on-demand ETH-SLM frame counts for CFM maintenance association end points (MEPs).

By default, the `show oam ethernet connectivity-fault-management mep-database` command displays CFM database information for MEPs in the specified CFM maintenance association within the specified CFM maintenance domain.

NOTE: At the router attached to the initiator MEP for a one-way session, or at the router attached to the receiver MEP for a two-way session, you can only display the ETH-SLM frame counts and not the MEP database details.

Action

- To display CFM database information (including ETH-SLM frame counts) for all MEPs in MA `ma1` within maintenance domain `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
ma1 maintenance-association ma1
```

- To display CFM database information (including ETH-SLM frame counts) only for the local MEP `201` in MA `ma1` within maintenance domain `md1`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
md2 maintenance-association ma2 local-mep 201
```

- To display CFM database information (including ETH-SLM frame counts) only for the remote MEP `302` in MA `ma3` within maintenance domain `md3`:

```
user@host> show oam ethernet connectivity-fault-management mep-database maintenance-domain
ma3 maintenance-association ma3 remote-mep 302
```

Meaning

The output displays ETH-SLM frame counts for MEPs within a particular maintenance domain, or for a specific local or remote MEP. For details about the output of this command and the descriptions of the output fields, see `show oam ethernet connectivity-fault-management mep-database`.

Displaying ETH-SLM Frame Counts for MEPs by Interface or Domain Level

IN THIS SECTION

- Purpose | 336
- Action | 337
- Meaning | 337

Purpose

Display on-demand ETH-SLM frame counts for CFM maintenance association end points (MEPs).

By default, the `show oam ethernet connectivity-fault-management interfaces` command displays CFM database information for MEPs attached to CFM-enabled Ethernet interfaces on the router or at a maintenance domain level. For Ethernet interfaces that support ETH-SLM, any frame counts are also displayed when you specify the `detail` or `extensive` command option.

NOTE: At the router attached to the initiator MEP, you can only display the ETH-SLM frame counts and not the MEP database details.

Action

- To display CFM database information (including ETH-SLM frame counts) for all MEPs attached to CFM-enabled Ethernet interfaces on the router:

```
user@host> show oam ethernet connectivity-fault-management interfaces detail
```

- To display CFM database information (including ETH-SLM frame counts) only for the MEPs attached to CFM-enabled router interface `ge-5/2/9.0`:

```
user@host> show oam ethernet connectivity-fault-management interfaces ge-5/2/9.0 detail
```

- To display CFM database information (including ETH-SLM frame counts) only for MEPs enclosed within CFM maintenance domains at level `6`:

```
user@host> show oam ethernet connectivity-fault-management interfaces level 6 detail
```

Meaning

The output displays ETH-SLM frame counts for MEPs for the specified interface. For details about the output of this command and the descriptions of the output fields, see `show oam ethernet connectivity-fault-management interfaces`.

Clearing ETH-SLM Statistics and Frame Counts

IN THIS SECTION

- Purpose | 338
- Action | 338

Purpose

Clear the on-demand ETH-SLM statistics and ETH-SLM frame counts.

By default, statistics and frame counts are deleted for all MEPs attached to CFM-enabled interfaces on the router. However, you can filter the scope of the command by specifying an interface name.

Action

- To clear the on-demand ETH-SLM statistics and ETH-SLM frame counts for all MEPs attached to CFM-enabled interfaces on the router:

```
user@host> clear oam ethernet connectivity-fault-management synthetic-loss-measurement
```

- To clear the on-demand ETH-SLM statistics and ETH-SLM frame counts only for MEPs attached to the logical interface `ge-0/5.9.0`:

```
user@host> clear oam ethernet connectivity-fault-management synthetic-loss-measurement
ge-0/5/9.0
```

Clearing Iterator Statistics

IN THIS SECTION

- Purpose | 338
- Action | 339

Purpose

Clear the existing iterator statistics and proactive ETH-SLM counters.

Multiple iterators can be associated with remote MEP. However, by default, only one result pertaining to one iterator profile can be cleared.

Action

- To clear the iterator statistics for remote MEP 1 and iterator profile i1 with MEPs belonging to the maintenance association ma1 within the maintenance domain default-1:

```
user@host> clear oam ethernet connectivity-fault-management sla-iterator-statistics sla-
iterator i1 maintenance-domain default-1 maintenance-association ma1 local-mep 1 remote-mep 1
```

- To clear the iterator statistics for remote MEP 1 and iterator profile i2 with MEPs belonging to the maintenance association ma1 within the maintenance domain default-1:

```
user@host> clear oam ethernet connectivity-fault-management sla-iterator-statistics sla-
iterator i2 maintenance-domain default-1 maintenance-association ma1 local-mep 1 remote-mep 1
```

RELATED DOCUMENTATION

clear oam ethernet connectivity-fault-management synthetic-loss-measurement

show oam ethernet connectivity-fault-management synthetic-loss-statistics

show oam ethernet connectivity-fault-management interfaces

show oam ethernet connectivity-fault-management mep-statistics

show oam ethernet connectivity-fault-management mep-database

Troubleshooting Failures with ETH-SLM

IN THIS SECTION

Problem | 339

Solution | 340

Problem

Description

The Ethernet synthetic loss measurement (ETH-SLM) application is not working properly for calculation of frame loss using synthetic frames instead of data traffic

Solution

Perform the following steps to analyze and debug any problems with the ETH-SLM functionality.

1. Ensure that ETH-SLM is configured (either proactive or on-demand) to initiate SLM frames. Verify the configuration settings.
2. Examine any failures that might have occurred in the CFM session for which the ETH-SLM feature is enabled. The CFM session must be in the up state for the ETH-SLM functionality to work correctly. Use the `show oam ethernet connectivity-fault-management mep-database maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* `local-mep` *`mep-id`* `remote-mep` *`remote-mep-id`* command to verify whether the CFM session is in the up state.
3. If the MEP sessions are active, use the appropriate show command to verify the ETH-SLM statistics and to analyze if ETH-SLM frames are transmitted or received.
4. If the transmission of ETH-SLM frames does not happen correctly after you attempt all of the preceding troubleshooting steps, enable the tracing operations for Ethernet CFM by including the `traceoptions` statement at the [`edit protocols oam ethernet connectivity-fault-management`] hierarchy level.

```
[edit protocols oam ethernet connectivity-fault-management]
traceoptions {
    file <filename> <files number  <match regular-expression microsecond-stamp>> <size size>
<world-readable | no-world-readable>;
    flag flag;
    no-remote-trace;
}
```

SEE ALSO

Ethernet Synthetic Loss Measurement Overview | 217

monitor ethernet synthetic-loss-measurement

RELATED DOCUMENTATION

ITU-T Y.1731 Ethernet Service OAM Overview | 204

Configure Ethernet Frame Loss Measurement | 268

Ethernet Alarm Indication

IN THIS SECTION

- Ethernet Alarm Indication Signal (ETH-AIS) Function Overview | 341
- Ethernet Alarm Indication Signal Overview | 346
- Configuring ETH-AIS on a CFM MEP | 347
- Configuring Alarm Indication Signal on ACX Series Routers | 353

Use the following to understand more about Ethernet alarm indication signal (ETH-AIS) and how to configure ETH-AIS on devices.

Ethernet Alarm Indication Signal (ETH-AIS) Function Overview

IN THIS SECTION

- Understanding ETH-AIS in a Maintenance Domain | 341
- Fault Detection in a Maintenance Domain | 342
- Terms Defined | 344

Ethernet alarm indication signal (ETH-AIS) function enables a service provider deploying an Ethernet service to determine whether a connectivity fault exists at the provider's domain level or at a level below. When the fault occurs at the provider's domain level, the service provider addresses the fault, and when the fault occurs at a level below, the provider can either ignore the fault or contact the relevant authorities to address the fault.

The following sections explain ETH-AIS, few use cases which determine when to generate and propagate ETH-AIS packets, and associated terms in detail:

Understanding ETH-AIS in a Maintenance Domain

ITU-T developed Y.1731 as a recommendation for Operation, Administration, and Maintenance (OAM) functions and mechanisms for Ethernet-based networks, including OAM functions such as ETH-AIS, Ethernet locked signal (ETH-LCK), Ethernet test signal (ETH-Test), Ethernet automatic protection switching (ETH-APS), Ethernet maintenance communication channel (ETH-MCC), Ethernet experimental

OAM (ETH-EXP), Ethernet vendor-specific OAM (ETH-VSP), and performance monitoring. For information about maintenance domain and related terms, see "Terms Defined" on page 344.

According to the Y.1731 standards, a server MEP is a combined function of the server layer termination function and the server Ethernet services layer adaptation function. The server MEP notifies the Ethernet services (ETH) layer MEPs when it detects a failure. The server layer termination function then runs the OAM mechanisms specific to the server layer and the alarms are suppressed at the server layer by ETH-AIS.

Note that ETH-AIS is not applicable to Spanning Tree Protocol (STP) networks.

ETH-AIS enables you to suppress alarms when a fault condition is detected. Using ETH-AIS, a service provider can differentiate between faults at different levels.

ETH-AIS provides many advantages that include:

- Service providers need not raise alarms if there are lower-level failures.
- Service providers can provide a refund to their subscribers or avail a refund from their Internet provider based on service unavailability.

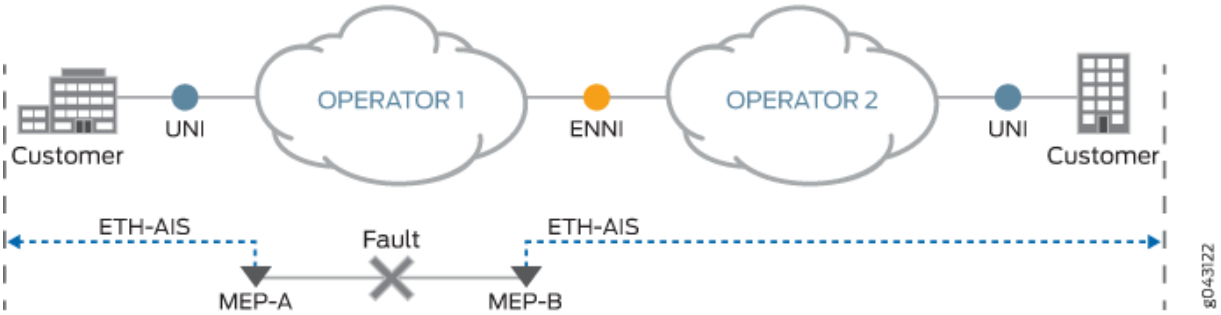
MX Series routers support ITU-T Y.1731 ETH-AIS to provide fault management for service providers who provide carrier Ethernet services using IEEE 802.1ag standard.

NOTE: MX Series Virtual Chassis does not support Ethernet alarm indication signal (ETH-AIS).

Fault Detection in a Maintenance Domain

In the scenario depicted in Figure 1 on page xyz, you have a service provider level and a customer level. Two service providers—*Operator-1* and *Operator-2*—are considered for illustration purposes. Assume that a fault occurs in Operator-1 maintenance domain-level that has MEP-A and MEP-B at its maintenance domain-level boundaries. To notify the faults to a network management system and to avoid notification of alarms from the customer level for the same fault, MEP-A and MEP-B transmit an alarm indication signal (AIS) on opposite directions, thereby signaling the higher levels and the Operator-2 network about the fault, so that the alarms are suppressed.

Signaling is achieved through transmission and propagation of AIS protocol data units (PDUs). You must enable AIS explicitly on all the MEPs at the service provider level. A MEP that is configured to issue frames with ETH-AIS information is generally at the server layer and continues to transmit periodic frames with ETH-AIS information until the defect condition is cleared. When a client MEP receives the ETH-AIS frames, it suppresses loss-of-continuity alarms associated with its peer MEPs.



Note that in the absence of AIS, a client MEP resumes generating loss-of-continuity alarms when it detects the loss-of-continuity defect conditions from its server layer.

For point-to-point Ethernet services layer connectivity, a MEP has only one peer MEP. Therefore, there is no ambiguity regarding the peer MEP for which the MEP should suppress alarms when it receives the ETH-AIS information.

For multipoint Ethernet services layer connectivity, a MEP that receives ETH-AIS information cannot determine the exact MEP that encountered the fault and, therefore, cannot isolate the exact peer MEP to suppress the alarms. To avoid this scenario, Y.1731 recommends suppressing alarms for all peer MEPs in the same domain level irrespective of connectivity status in a multipoint Ethernet services layer connectivity setup.

Table 19 on page 343 lists the operational mode commands that you can use in a maintenance domain to check the various parameters pertaining to a MEP.

Table 19: Operational Mode Commands

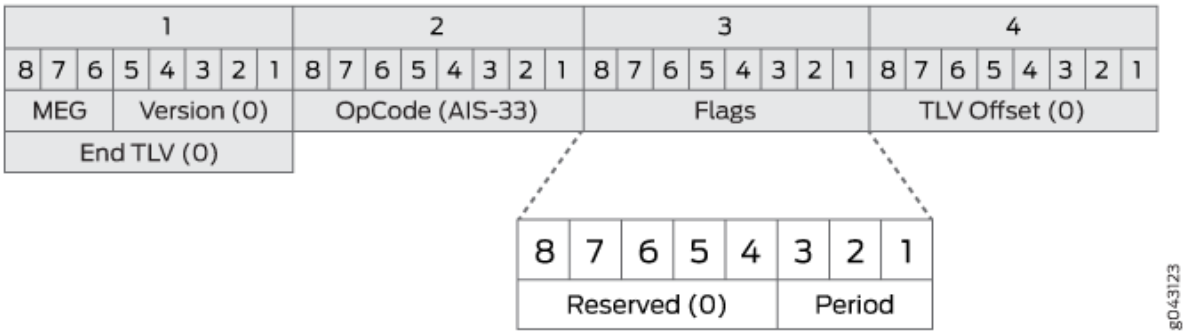
To Check	Operational Mode Commands
Whether the AIS configuration is configured correctly on a CFM MEP.	`show protocols oam ethernet connectivity-fault-management action-profile`
Statistics of AIS frames.	`show oam ethernet connectivity-fault-management interfaces detail` `show oam ethernet connectivity-fault-management mep-statistics maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* `remote-mep` *`mep-id`* `local-mep` *`mep-id`*

Table 19: Operational Mode Commands *(Continued)*

To Check	Operational Mode Commands
Whether any event has occurred that triggered AIS.	`show oam ethernet connectivity-fault-management mep-database maintenance-domain` *`md-name`* `maintenance-association` *`ma-name`* `remote-mep` *`mep-id`* `local-mep` *`mep-id`*
Status of CFM sessions for faults that trigger AIS on the MEP.	`show oam ethernet connectivity-fault-management interfaces detail`

Terms Defined

- AIS transmission—A MEP upon detecting a defect condition transmits AIS frames in a direction opposite to its peer MEPs. The periodicity of AIS frames transmission is on the basis of the AIS transmission period. An AIS transmission period of 1 second is recommended. The first AIS frame must always be transmitted immediately following the detection of a defect condition.
- AIS reception—Upon receiving an AIS frame, a MEP examines it to ensure that the frame's maintenance domain level is the same as its own maintenance domain level. The *period* field in the frame indicates the period at which the AIS frames can be expected. When a MEP receives an AIS frame, it detects the defect condition. After detection, when no AIS frames are received within an interval of 3.5 times—the AIS transmission period indicated in the AIS frames received—the MEP clears the AIS defect condition. When the AIS condition is cleared and defects still exist, then the MEPs continue to report alarms.
- AIS PDU format—The fields of the AIS PDU format are:



1. MEG Level—Also called the maintenance domain level, it is a 3-bit field that is used to carry the maintenance domain level of the client MEG.

2. Version—Value is always 0.
3. OpCode—Value for this PDU type is AIS (33).
4. Flags—The first five bits are reserved and are set to 0. The 3-bit information element carried in the three least significant bits are referred to as the period that contains the value of AIS transmission periodicity as illustrated in Table 20 on page 345:

   Table 20: AIS Transmission Periodicity

Flags [3:1]	Period Value	Comments
000-011	Invalid value	Invalid value for AIS
100	1s	1 frame per second
101	Invalid value	Invalid value for AIS
110	1 min	1 frame per minute
111	Invalid value	Invalid value for AIS

5. TLV offset—Set to 0.
6. End TLV—All-zeroes octet value.

- Server layer and client layer—These layers are part of the ITU-T Recommendation G.805 transport network functional model. This model is based on the concept of layering within a transport network. A transport network is divided into several independent transport layer networks that have a client-server association between adjacent layer networks.
- Maintenance domain—To enable connectivity fault management (CFM) on an Ethernet interface, maintenance domains, maintenance associations, and maintenance end points (MEPs) are created and configured in a network. You can configure up to eight maintenance domain levels in a network. Each maintenance domain level is a part of the network where the connectivity issues can be monitored and corrected. Provider domain and customer domain are some examples for maintenance domains. Each maintenance domain has a maintenance association. Each maintenance association includes MEPs and maintenance intermediate points (MIPs) in that domain. The MEPs are located at the boundary of the domain and the MIPs are located within the domain. MEPs generate and transmit continuity check messages (CCMs) at configured intervals to the entire maintenance association to check the connectivity in the network.

- Ethernet services (ETH) layer—A layer in the metro Ethernet network model, where this layer is responsible for the OAM services that are required to support the Ethernet services in the network.

SEE ALSO

> show oam ethernet connectivity-fault-management interfaces
>
> *show oam ethernet connectivity-fault-management mep-statistics*

Ethernet Alarm Indication Signal Overview

ACX Series routers support ITU-T Y.1731 Ethernet Alarm Indication Signal function (ETH-AIS) to provide fault management for service providers. ETH-AIS enables you to suppress alarms when a fault condition is detected. Using ETH-AIS, an administrator can differentiate between faults at customer level or faults at provider level.

The advantage of ETH-AIS is:

- Customers need not raise alarms due to lower level failures.
- Customers can get refund based on service unavailability.

When a fault condition is detected, a maintenance end point (MEP) generates ETH-AIS packets to the configured client levels for a specified duration until the fault condition is cleared. Any MEP configured to generate ETH-AIS packets signals to a level higher than its own. A MEP receiving ETH-AIS recognizes that the fault is at a lower level and then suppresses alarms at current level.

ACX Series routers support ETH-AIS PDU generation for server MEPs based on the following defect conditions:

- Loss of connectivity (physical link loss detection)
- Layer 2 circuit or Layer 2 VPN down

Alarm indication signaling is done through the transmission and propagation of ETH-AIS PDUs. ETH-AIS should be enabled on MEPs. A MEP which is configured to issue packets with ETH-AIS information is generally of server layer and continues to transmit periodic packets with ETH-AIS information until the defect condition is cleared. CFM MEPs, upon receiving ETH-AIS PDUs, suppresses loss of continuity alarms associated with its peer MEPs. A MEP resumes loss of continuity alarm generation upon detecting loss of continuity defect conditions in the absence of an ETH-AIS condition.

For point-to-point Ethernet connectivity, a MEP has only a single peer MEP. Therefore, a MEP suppress alarms on its peer MEP when it receives the ETH-AIS information.

For multi-point Ethernet connectivity, a MEP which receives ETH-AIS information cannot determine the exact MEP encountered a fault condition and therefore it will not be able to isolate the exact peer MEP

for alarm suppression. ITU-T Y.1731 recommends suppressing alarms for all peer MEPs irrespective of the connectivity status.

AIS transmission—A MEP upon detecting a defect condition transmits ETH-AIS PDUs in a direction opposite to its peer MEPs. The transmission of ETH-AIS PDUs is based on a configured ETH-AIS transmission period. An ETH-AIS transmission period of 1 second is recommended. The first ETH-AIS PDU must be transmitted immediately following the detection of a defect condition.

AIS reception—A MEP upon receiving ETH-AIS PDUs examines it to ensure that its maintenance domain (MD) level corresponds to the same MD level. Upon receiving an ETH-AIS PDU, the MEP detects a defect condition. Following the detection of a defect condition, if there are no ETH-AIS PDUs received within an interval of 3.5 times the ETH-AIS transmission period indicated in the ETH-AIS PDUs received earlier, the MEP clears the defect condition. After the fault condition is cleared, MEPs continue to report alarms.

NOTE: ACX Series routers do not support ITU-T Y.1731 ETH-AIS for layer 2 services (bridging).

The following are the limitations for server MEP

- Triggering of ETH-AIS messages over services (Layer 2 circuit and Layer 2 VPN) by the link-loss server MEP is done on a best-effort manner. This is because the transmission of ETH-AIS messages is independent of the service status and there is no guarantee for delivering the ETH-AIS messages before service goes down.
- Pseudowire protection with CFM-MEP session is not monitored by the server-MEP because an entity to monitor pseudowire protection already exists for the service (Layer 2 circuit and Layer 2 VPN).

SEE ALSO

show oam ethernet connectivity-fault-management mep-statistics

Configuring ETH-AIS on a CFM MEP

IN THIS SECTION

- Configuring an Action Profile | **348**
- Configuring an Action to Be Taken When an AIS Alarm Is Detected | **350**
- Attaching the Action Profile to a CFM MEP | **351**

MX Series routers support ITU-T Y.1731 Ethernet alarm indication signal (ETH-AIS) function to provide fault management for service providers. ETH-AIS enables the service provider to suppress alarms when a fault condition is detected.

The following points are to be noted when ETH-AIS is configured in a maintenance domain:

- Transmitting or receiving of AIS on a MEP does not override the `lowest-priority-defect` statement configured at the `[edit protocols oam ethernet connectivity-fault-management maintenance-domain` *`domain-name`* `maintenance-association` *`ma-name`* `mep` *`mep-id`*] hierarchy level. Therefore, alarms are reported according to the defect priority configured.
- Alarms are reported even when the higher domain levels exchange CCMs at a faster rate than the lower domain levels.
- Maintenance association intermediate point (MIP) is transparent to ETH-AIS frames—that is, the MIPs do not perform any action in response to ETH-AIS frames.
- When the service provider requests the MEP to generate an AIS for a lower level or for the same level, the request is rejected.
- AIS generation is stopped when the MEP clears the remote MEP within the maintenance association.
- When the `auto-discovery` statement is enabled for a MEP, the remote MEP information is cleared after the configured hold interval expires.

The following tasks explain how to enable ETH-AIS in a maintenance domain, configure an action to be taken when a defect is detected, and to attach the action profile to a CFM MEP:

Configuring an Action Profile

To configure an action profile for ETH-AIS:

1. Go to the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management
```

2. Configure an action profile to use when one or more remote MEPs are down.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# edit action-profile action-profile-name
```

3. Configure an event that needs to be monitored.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name]
user@host# edit event
```

4. Configure the defect condition that generates an alarm indication signal.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
event]
user@host# edit ais-trigger-condition
```

5. Configure the `adjacency-loss` statement to inform the operator when the physical connectivity is lost between the peer MEPs.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
event ais-trigger-condition]
user@host# set adjacency-loss
```

6. Configure the `all-defects` statement to inform the operator that all possible defects must be considered to raise the alarm indication signal.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
event ais-trigger-condition]
user@host# set all-defects
```

7. Configure the `cross-connect-ccm` statement to inform the operator when cross-connect continuity check messages (CCMs) are received by the MEP and to raise an alarm indication signal in response.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
event ais-trigger-condition]
user@host# set cross-connect-ccm
```

8. Configure the `erroneous-ccm` statement to inform the operator when CCMs with unexpected MEP ID or maintenance domain level are received by the MEP and an AIS alarm is raised in response.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
event ais-trigger-condition]
user@host# set erroneous-ccm
```

9. Configure the `receive-ais` statement to inform the operator that an AIS message has been received from the peer MEP in its own maintenance level.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
event ais-trigger-condition]
user@host# set receive-ais
```

Configuring an Action to Be Taken When an AIS Alarm Is Detected

Configure an action to be taken when an AIS alarm is detected.

1. Go to the `[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name action]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management action-profile action-
profile-name action
```

2. Configure the `log-and-generate-ais` statement to log the event that generated the AIS message.

```
[edit protocols oam ethernet connectivity-fault-management action-profile action-profile-name
action]
user@host# edit log-and-generate-ais
```

3. Configure the interval between AIS messages that are to be received by the MEP as 1 minute or 1 second.

```
[edit protocols oam ethernet connectivity-fault-management  action-profile action-profile-
name action log-and-generate-ais]
user@host# set interval (1m | 1s)
```

4. Configure the server maintenance domain level range of the MEP from 1 through 7.

```
[edit protocols oam ethernet connectivity-fault-management  action-profile action-profile-
name action log-and-generate-ais]
user@host# set level level
```

5. Configure the 802.1p priority of the AIS packet from 1 through 7.

```
[edit protocols oam ethernet connectivity-fault-management  action-profile action-profile-name action log-and-generate-ais]
user@host# set priority level
```

Attaching the Action Profile to a CFM MEP

After configuring an event and an action to be monitored in an action profile, you must attach the action profile to a CFM MEP.

1. Go to the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management
```

2. Configure the maintenance domain with a name.

```
[edit protocols oam ethernet connectivity-fault-management]
user@host# edit maintenance-domain md-name
```

3. Configure the maintenance domain with a client maintenance entity group (MEG) level or maintenance association level—the level which the client layer maintenance association intermediate point (MIPs) and the MEPs exist—from 0 through 7.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name]
user@host# edit level level
```

NOTE: You cannot configure a maintenance domain level that is lower than or equal to the maintenance association level that it is associated with.

4. Configure the maintenance association.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name]
user@host# edit maintenance-association ma-name
```

5. Configure the continuity check that is performed on all the MEPs in a domain level by sending CCMs with an interval between two CCMs—100 miiliseconds, 10 milliseconds, 1 second, 10

seconds, 1 minute, or 10 minutes—and the number of CCMs that are to be lost before marking a MEP as down.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name]
user@host# set continuity-check interval (100ms | 10m | 10ms | 1m | 1s)
user@host# set continuity-check loss-threshold value
```

6. Configure the MEP with an identifier from 1 through 8192.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name]
user@host# set mep mep-id
```

7. Attach the configured action profile to the MEP.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id]
user@host# set action-profile action-profile-name
```

8. Configure the interface of the MEP over which the CCMs are transmitted.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id]
user@host# set interface interface-name
```

9. Configure the direction for the CCMs to travel to the next MEP as up or down.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id]
user@host# set direction (down | up)
```

10. Configure the 802.1p priority for the CCMs and link-trace packet from 0 through 7.

```
[edit protocols oam ethernet connectivity-fault-management maintenance-domain md-name
maintenance-association ma-name mep mep-id]
user@host# set priority priority-value
```

Configuring Alarm Indication Signal on ACX Series Routers

ACX Series routers support ITU-T Y.1731 Ethernet Alarm Indication Signal function (ETH-AIS) to provide fault management for service providers. ETH-AIS enables you to suppress alarms when a fault condition is detected.

To support ETH-AIS transmission, the following configuration information is required by a CFM MEP:

- Client Maintenance Entity Group level—Maintenance Entity Group (MEG) level at which the immediate client layer Maintenance Domain Intermediate Points (MIPs) and Maintenance Association End Points (MEPs) exist.
- ETH-AIS transmission period—Determines the ETH-AIS PDU transmission interval.
- Priority—Determines the priority of packets with ETH-AIS information. This is optional.

To configure ETH-AIS in CFM MEP, you need to:

- Configure an action profile with ETH-AIS action
- Attach the action profile to the CFM MEP

To configure an action profile with ETH-AIS action, include the following statements at the [edit protocols oam ethernet connectivity-fault-management] hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management]
action-profile action-profile-name {
    event {
        adjacency-loss;
        all-defects;
        cross-connect-ccm;
        errored-ccm;
        receive-ais;
    }
    action {
        log-and-generate-ais {
            level [1-7];
            interval 1s | 1m ;
            priority [0-7];
        }
        log-ais;
    }
}
```

To attach an action profile to a CFM MEP, include the following statements at the [edit protocols oam ethernet connectivity-fault-management] hierarchy level:

```
maintenance-domain maintenenace-domain-name {
level level-number;
    maintenance-association maintenance-domain-name {
        continuity-check {
            interval 1s;
            loss-threshold 3;
        }
        mep mep-id {
            interface interface-name;
            direction up | down;
            priority priority-value;
            action-profile action-profile-name;
        }
    }
}
```

NOTE: You cannot configure a maintenance domain level that is lower than or equal to the level that it is associated with.

To support ETH-AIS transmission, the following configuration information required by a server MEP:

- Server MEP definition—Defines the association of server MEP identifier to the server layer.
  - For Layer 2 circuit and Layer 2 VPN, the logical interface connected to a customer network (UNI) would be the identifier for the server layer that needs to be monitored by the server MEP.
  - For physical link loss detection, the physical interface under Ethernet protocol would be the identifier for the server layer that needs to be monitored by the server MEP.
- Association of server MEP defect—Defines the association of server MEP defects to ETH-AIS action.
- Association action profile and server MEP—Defines the binding of server MEP and action profile.

To configure ETH-AIS in server MEP, you need to:

- Create an action profile with ETH-AIS action for server MEP defects.
- Attach the action profile to a server MEP

To create an action profile, include the following statements at the [edit protocols oam ethernet connectivity-fault-management] hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management]
action-profile action-profile-name {
    event {
        server-mep-defects {
            link-loss-defect;
            l2circuit-defect;
            l2vpn-defect;
        }
    }
    action {
        log-and-generate-ais {
            level 1…n;
            interval 1 second | 1 minute;
            priority dot1p [range 0-7];
        }
    }
}
```

To attach an action profile to a server MEP, include the following statement at the [edit protocols oam ethernet connectivity-fault-management] hierarchy level:

```
[edit protocols oam ethernet connectivity-fault-management]
server-mep  mep-identifier {
    protocol l2circuit | l2vpn | ethernet {
        interface interface-name;
    }
    action-profile action-profile-name;
}
```

Inline Transmission Mode

IN THIS SECTION

- Enabling Inline Transmission of Continuity Check Messages for Maximum Scaling | 356
- Enabling Inline Transmission of Link Fault Management Keepalives for Maximum Scaling | 357
- Enabling Inline Mode Of Performance Monitoring To Achieve Maximum Scaling | 361
- Supported Inline CCM and Inline PM Scaling Values | 364

Use this topic to understand what inline transmission is and how to enable it for maximum scaling for CFM, LFM, and performance monitoring functions.

Enabling Inline Transmission of Continuity Check Messages for Maximum Scaling

Scaling is the ability of a system to handle increasing amounts of work and to continue to function well. Scaling can refer to increasing capacity and the ability to handle increasing workload, number of subscribers or sessions, hardware components, and so on. Continuity check protocol is used for fault detection within a maintenance association. The maintenance association end points (MEPs) send continuity check messages (CCMs) periodically. The time between the transmissions of CCMs is known as the interval. The receiving MEP maintains a database of all MEPs in the maintenance association.

By default, CCMs are transmitted by the CPU of a line card, such as a Modular Port Concentrator (MPC). If the duration between transmissions of CCMs is low or if the CCMs for a specific line card scale, then we recommend that you delegate transmission of CCMs to the forwarding ASIC (that is, to the hardware) by enabling inline transmission of CCMs. Inline transmission of CCMs is also known as inline keepalives or Inline-KA. Inline transmission enables the system to handle more connectivity fault management (CFM) sessions per line card. By enabling inline transmission of CCMs, you can achieve maximum scaling of CCMs.

To enable inline transmission of CCMs, perform the following steps:

1. In configuration mode, go to the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management performance-monitoring
```

2. Delegate transmission of CCMs to hardware by enabling hardware-assisted keepalives.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# set hardware-assisted-keepalives enable
```

NOTE: Inline transmission of CCMs is not enabled when there is a CFM session already established. To enable inline transmission, you must first deactivate the CFM session using the `deactivate` command and then reactivate the CFM session using the `activate` command.

To disable inline transmission, use the `hardware-assisted-keepalives disable` statement. After disabling inline transmission, you must reboot the router for the changes to take effect.

SEE ALSO

Configure Connectivity Fault Management for Interoperability During Unified In-Service Software Upgrades | 52

Enabling Inline Transmission of Link Fault Management Keepalives for Maximum Scaling

Scaling is the ability of a system to handle increasing amounts of work and to continue to function well. Scaling can refer to increasing capacity and the ability to handle increasing workload, number of subscribers or sessions, hardware components, and so on.

By default, LFM keepalive packets are transmitted by the periodic packet management `ppm` process on the line-card. You can delegate transmission of LFM keepalive packets to the forwarding ASIC (that is, to the hardware) by enabling inline transmission. Inline transmission of LFM keepalives is also known as inline keepalives or Inline-KA. By enabling inline transmission of LFM keepalive packets, you can achieve maximum scaling of keepalive packets, reduction of the load on the `ppm` process, and support LFM in-service software upgrade (ISSU) for non-juniper peers (for a keepalive interval of 1 second).

NOTE: Do not enable or disable inline transmission of LFM when an LFM session is already established. To enable or disable inline transmission, you must first deactivate the existing established LFM session using the `deactivate` command, and then reactivate the LFM session using the `activate` command after enabling or disabling inline LFM.

Before you enable inline transmission of LFM keepalive packets, complete the following tasks:

- Verify if any LFM session is online and active. To verify if any existing or established LFM session is online and active, issue the following command:

```
user@host> show oam ethernet link-fault-management detail
Oct 18 02:04:17
  Interface: ge-0/0/0
    Status: Running, Discovery state: Active Send Local
    Transmit interval: 1000ms, PDU threshold: 3 frames, Hold time: 0ms
    Peer address: 00:00:00:00:00:00
    Flags:0x8
    OAM receive statistics:
      Information: 0, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 0, Organization specific: 0
    OAM flags receive statistics:
      Critical event: 0, Dying gasp: 0, Link fault: 0
    OAM transmit statistics:
      Information: 28, Event: 0, Variable request: 0, Variable response: 0  = after waiting
for a while count increased by 15
      Loopback control: 0, Organization specific: 0
    OAM received symbol error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame period error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame seconds error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM transmitted symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM current symbol error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM transmitted frame error event information:
      Events: 0, Window: 0, Threshold: 1
      Errors in period: 0, Total errors: 0
    OAM current frame error event information:
      Events: 0, Window: 0, Threshold: 1
```

```
      Errors in period: 0, Total errors: 0
    Loopback tracking: Disabled, Loop status: Unknown
    Detect LOC: Disabled, LOC status: Unknown
```

The OAM transmit statistics reflect that the `ppm` process is handling the transmission of LFM keepalive packets.

- Deactivate the LFM session so that you can enable inline LFM mode. To deactivate the LFM session, issue the following command:

```
[edit]
user@host # deactivate protocols oam ethernet link-fault-management interface interface-name
```

- Commit the configuration. To commit the configuration, issue the following command:

```
[edit]
user@host # commit
```

To enable inline transmission of LFM keepalive packets, perform the following steps:

1. In configuration mode, go to the `[edit protocols oam ethernet link-fault-management]` hierarchy level.

```
[edit]
user@host# edit protocols oam ethernet link-fault-management
```

2. Delegate transmission of LFM keepalive packets to hardware by enabling hardware-assisted keepalives.

```
[edit protocols oam ethernet link-fault-management]
user@host# set hardware-assisted-keepalives
```

3. Commit the configuration.

```
[edit]
user@host # commit
```

4. Re-activate the LFM session as follows:

```
[edit]
user@host # activate protocols oam ethernet link-fault-management interface interface-name
```

5. Commit the configuration.

```
[edit]
user@host # commit
```

6. Verify that the trasnmission of LFM keepalive packets is delegated from the ppm process to the hardware. To verify that you have enabled inline transmission, issue the following command:

```
user@host> show oam ethernet link-fault-management detail
Oct 18 02:05:05
  Interface: ge-0/0/0
    Status: Running, Discovery state: Active Send Local
    Transmit interval: 1000ms, PDU threshold: 3 frames, Hold time: 0ms
    Peer address: 00:00:00:00:00:00
    Flags:0x8
    OAM receive statistics:
      Information: 0, Event: 0, Variable request: 0, Variable response: 0
      Loopback control: 0, Organization specific: 0
    OAM flags receive statistics:
      Critical event: 0, Dying gasp: 0, Link fault: 0
    OAM transmit statistics:
      Information: 1, Event: 0, Variable request: 0, Variable response: 0  = even after 10
seconds count is still 1
      Loopback control: 0, Organization specific: 0
    OAM received symbol error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame period error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
    OAM received frame seconds error event information:
      Events: 0, Window: 0, Threshold: 0
      Errors in period: 0, Total errors: 0
```

```
OAM transmitted symbol error event information:
  Events: 0, Window: 0, Threshold: 1
  Errors in period: 0, Total errors: 0
OAM current symbol error event information:
  Events: 0, Window: 0, Threshold: 1
  Errors in period: 0, Total errors: 0
OAM transmitted frame error event information:
  Events: 0, Window: 0, Threshold: 1
  Errors in period: 0, Total errors: 0
OAM current frame error event information:
  Events: 0, Window: 0, Threshold: 1
  Errors in period: 0, Total errors: 0
Loopback tracking: Disabled, Loop status: Unknown
Detect LOC: Disabled, LOC status: Unknown
```

The OAM transmit statistics are not updated. When you enable inline transmission of LFM keepalive packets, the OAM transmit statistics are not updated.

To disable inline LFM, verify if any existing established LFM session is online and active. Deactivate the LFM session and commit. Disable inline LFM by deleting the `hardware-assisted-keepalives` statement and commit. Then, reactivate LFM session and commit the configuration.

SEE ALSO

hardware-assisted-keepalives (lfm)

Enabling Inline Mode Of Performance Monitoring To Achieve Maximum Scaling

Performance monitoring is useful for studying the traffic pattern in a network over a period of time. It helps to identify network problems before you are impacted by network defects.

By default, performance monitoring packets are handled by the CPU of a line-card, such as Modular Port Concentrator (MPC). Enabling inline mode of performance monitoring delegates the processing of the protocol data units (PDUs) to the forwarding ASIC (that is, to the hardware). By enabling inline mode of performance monitoring, the load on the CPU of the line-card is reduced and you can configure an increased number of performance monitoring sessions and achieve maximum scaling for service OAM performance monitoring sessions. On MX Series routers, you can configure inline mode of performance monitoring only if the network services mode on the router is configured to `enhanced-ip` and enhanced connectivity fault management (`enhanced-cfm-mode`)) is configured.

By enabling inline mode of performance monitoring, you can achieve maximum scaling for performance monitoring sessions. To achieve maximum scaling for performance monitoring sessions, you must enable scaling of continuity check messages (CCMs) sessions. To enable scaling of CCM sessions, enable inline

transmission of continuity check messages. For more information on inline transmission of continuity check messages, see "Enabling Inline Transmission of Continuity Check Messages for Maximum Scaling" on page 356. To view the supported scaling values for CCM and PM, see "Supported Inline CCM and Inline PM Scaling Values" on page 364.

Inline mode of performance monitoring is supported only for proactive mode of frame delay measurement (Two-way Delay Measurements) and synthetic loss measurements (SLM)sessions. Performance monitoring functions configured using the iterator profile (CFM)are referred to as proactive performance monitoring. Inline mode of performance monitoring for frame loss measurement using service frames (LM) is not supported.

> **NOTE**: MPC3E (MX-MPC3E-3D) and MPC4E (MPC4E-3D-32XGE-SFPP and MPC4E-3D-2CGE-8XGE) do not support inline mode of performance monitoring. User-defined Data TLV is not supported if you have configured inline mode of performance monitoring. Also, only 12 history records per PM sessions are supported.
>
> We recommend that you enable inline mode of performance monitoring before you configure the performance monitoring sessions as the change may interfere with the existing performance monitoring sessions.

To enable inline mode of performance monitoring, perform the following steps:

1. In configuration mode, go to the `[edit chassis]` hierarchy level and configure the network services mode of the router. The network service mode of the router must be configured as `enhanced ip` to enable enhanced connectivity fault management (CFM) mode.

   > **NOTE**: If the network services mode is not `enhanced-ip`, and you have enabled enhanced CFM, the following warning message is displayed:
   >
   > `[edit protocols oam ethernet] 'connectivity-fault-management' enhanced ip is not effective please configure enhanced ip and give router reboot`

   ```
   [edit chassis]
   user@host# set network-services enhanced-ip
   ```

2. In configuration mode, go to the `[edit protocols oam ethernet connectivity-fault-management]` hierarchy level and enable enhanced connectivity fault management mode by using the `enhanced-cfm-mode` option.

   ```
   [edit]
   user@host# set protocols oam ethernet connectivity-fault-management enhanced-cfm-mode
   ```

3. In configuration mode, go to the `[edit protocols oam ethernet connectivity-fault-management performance-monitoring]` hierarchy level. Configure the enhanced iterator profile by using the `enhanced-sla-iterator` option and specify the measurement interval by using the `measurement-interval` option.

```
[edit]
user@host# edit protocols oam ethernet connectivity-fault-management performance-monitoring
enhanced-sla-iterator measurement-interval value
```

4. Enable inline performance monitoring.

> **NOTE**: You can enable inline mode of performance monitoring for both the originator and the responder of the service OAM performance monitoring sessions by using the `hardware-assisted-pm` command.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# set hardware-assisted-pm
```

5. (Optional) Enable inline transmission of CCMs to enable better scaling if inline transmission of CCMs is not automatically enabled.

> **NOTE**: You can achieve better scaling if both inline performance monitoring and inline transmission of CCMs is enabled.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# set hardware-assisted-keepalives enable
```

6. Commit the configuration.

```
[edit protocols oam ethernet connectivity-fault-management performance-monitoring]
user@host# commit
```

SEE ALSO

Enabling Enhanced Connectivity Fault Management Mode | 50

Network Services Mode Overview

hardware-assisted-pm

Supported Inline CCM and Inline PM Scaling Values

This topic lists the scaling values for inline mode of performance monitoring and inline transmission of continuity check messages. The scaling values are based on the different cycle-time interval values. Each table lists the maximum number of connectivity fault management (CFM) sessions and performance monitoring (PM) sessions per line card and per chassis when you configure inline CCM, enhanced CFM, and enhanced PM by using the `hardware-assisted-keepalives`, `enhanced-cfm-mode`, and `hardware-assisted-pm` options.

NOTE: The scaling values do not consider the load from other protocols in the system and so the actual realized scaling values for line card and chassis vary depending on other protocol configurations and scaling in the system. We recommend that you configure DDoS for CFM. Limit the number of CFM packets, that are sent to the CPU of the line card, to 3000. Limiting the number of packets safeguards the CPU from scaled CFM configurations of various CFM protocol events.

Table 21 on page 364 lists the maximum number of connectivity fault management (CFM) sessions and performance monitoring (PM) sessions per line card and per chassis when you configure both the CCM interval and the PM interval as 1 second.

Table 21: Scaling Values for CFM and PM (CCM Interval: 1 sec and PM Interval: 1 sec)

CFM Line Card Scale	PM Line Card Scale	CFM Chassis Scale	PM Chassis Scale
4000	4500	16000	16000
6000	3750	16000	16000
7000	3375	16000	16000
8000	3000	16000	16000

Table 22 on page 365 lists the maximum number of connectivity fault management (CFM) sessions and performance monitoring (PM) sessions per line card and per chassis when you configure the CCM interval as 1 second and the PM interval as 100 milliseconds.

Table 22: Scaling Values for CFM and PM (CCM Interval: 1 sec and PM interval: 100 ms)

CFM Line Card Scale	PM Line Card Scale	CFM Chassis Scale	PM Chassis Scale
4000	450	12000	4000
6000	375	12000	4000
7000	337	12000	4000
8000	300	12000	4000

Table 23 on page 365 lists the maximum number of connectivity fault management (CFM) sessions and performance monitoring (PM) sessions per line card and per chassis when you configure the CCM interval as 100 milliseconds and the PM interval as 1 second.

Table 23: Scaling Values for CFM and PM (CCM Interval: 100 ms and PM interval: 1 sec)

CFM Line Card Scale	PM Line Card Scale	CFM Chassis Scale	PM Chassis Scale
4000	3000	8000	6000
3000	3750	8000	6000
2000	4500	8000	6000
1000	4500	8000	6000

Table 24 on page 365 lists the maximum number of connectivity fault management (CFM) sessions and performance monitoring (PM) sessions per line card and per chassis when you configure both the CCM interval and the PM interval as 100 milliseconds.

Table 24: Scaling Values for CFM and PM (CCM Interval: 100 ms and PM interval: 100 ms)

CFM Line Card Scale	PM Line Card Scale	CFM Chassis Scale	PM Chassis Scale
4000	300	8000	3000

Table 24: Scaling Values for CFM and PM (CCM Interval: 100 ms and PM interval: 100 ms) *(Continued)*

CFM Line Card Scale	PM Line Card Scale	CFM Chassis Scale	PM Chassis Scale
3000	375	8000	3000
2000	450	8000	3000
1000	450	8000	3000

SEE ALSO

enhanced-cfm-mode

hardware-assisted-pm

RELATED DOCUMENTATION

Configure Continuity Check Messages | **107**

Introduction to OAM Link Fault Management (LFM) | **132**

3

PART

Network Monitoring by using SNMP

SNMP Architecture and SNMP MIBs Overview | 369

Understand SNMP Implementation in Junos OS | 371

Configure SNMP in Junos OS | 379

Configure Options on Managed Devices for Better SNMP Response Time | 399

Enterprise Specific Utility MIB to Enhance SNMP Coverage | 402

Optimize the Network Management System Configuration for the Best Results | 405

Interfaces to Accept SNMP Requests | 407

Configure SNMP for Routing Instances | 410

Configure SNMP Remote Operations | 432

SNMP Traps | 454

SNMP Traps Supported by Junos OS | 467

Trace SNMP Activity | 514

Access Privileges for an SNMP Group | 524

Configure Local Engine ID on SNMPv3 | 532

Configure SNMPv3 | 533

Configure SNMPv3 Authentication Type and Encryption Type | 540

SNMPv3 Traps | 543

SNMPv3 Informs | 550

SNMP Communities | 558

MIB Views | 572

SNMP MIBs Supported by Junos OS and Junos OS Evolved | 575

Junos OS SNMP FAQs | 656

SNMP Architecture and SNMP MIBs Overview

IN THIS SECTION

- SNMP Architecture | 369

SNMP Architecture

A typical SNMP implementation includes three components:

- Network Management System (NMS)—A combination of hardware (devices) and software (the SNMP manager) used to monitor and administer a network. The manager polls the devices on your network as you specify for information about network connectivity, activity, and events.
- Managed device—A managed device (also called a network element) is any device on a network managed by the NMS. Routers and switches are common examples of managed devices.
- SNMP agent—The SNMP agent is the SNMP process that resides on the managed device and communicates with the NMS. The SNMP agent exchanges network management information with the SNMP manager software running on an 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.

This topic contains the following sections:

SNMP MIBs

You can store SNMP data in a highly structured, hierarchical format known as a Management Information Base (MIB). A MIB defines managed objects in a network device.

The MIB structure is based on a tree structure and 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. For more information, see Table 25 on page 370.

Table 25: Standard and Enterprise-specific MIBs

Standard MIBs	Enterprise-specific MIBs
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.	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 standard supported MIBs, see Standard SNMP MIBs Supported by Junos OS.	For a list of Juniper Networks enterprise-specific supported MIBs, see Enterprise-Specific SNMP MIBs Supported by Junos OS.

SNMP Manager and Agent Authentication and Communication

SNMP uses a basic form of authentication called community strings to control access between a manager and remote agents. Community strings are administrative names used to group collections of devices (and the agents running on them) into common management domains. If a manager and an agent share the same community, they can talk to one another. Many people associate SNMP community strings with passwords and keys because the jobs they do are similar. As a result, SNMP communities are traditionally referred to as strings.

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.

SNMP Traps and Informs

Routers can send notifications to SNMP managers when significant events occur on a network device, most often errors or failures. You can send SNMP notifications as traps or inform requests.

SNMP traps are unconfirmed notifications and SNMP informs are confirmed notifications.

SNMP traps are either standard or enterprise-specific. For more information, see Table 26 on page 371.

Table 26: Standard and Enterprise-specific Traps

Standard Traps	Enterprise-specific Traps
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.	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 standard traps supported by the Junos OS, see Standard SNMP Traps Supported on Devices Running Junos OS.	For more information about enterprise-specific traps supported by the Junos OS, see Enterprise-Specific SNMP Traps Supported by Junos OS. For information about system logging severity levels for SNMP traps, see No Link Title.

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 No Link Title.

Understand SNMP Implementation in Junos OS

IN THIS SECTION

- Loading MIB Files to a Network Management System | 375
- Understand the Integrated Local Management Interface | 378

SNMP on Junos OS

On Junos OS, SNMP uses both standard (developed by the IETF and documented in RFCs) and Juniper Networks enterprise-specific MIBs.

NOTE: By default, SNMP is not enabled on devices running Junos OS.

In Junos OS, the processes that maintain the SNMP management data include the following:

- A master SNMP agent resides on the managed device and is managed by the NMS, or host.

  The Junos OS SNMP agent software consists of an SNMP primary agent (known as the SNMP process, or snmpd). It resides on the managed device and is managed by the NMS or host.

- Various subagents that reside on different modules of Junos OS, such as the Routing Engine. The master SNMP agent delegates all SNMP requests to the subagents. Each subagent is responsible for the support of a specific set of MIBs.

- Junos OS processes that share data with the subagents when polled for SNMP data (for example, interface-related MIBs).

The community string is the first level of management authentication implemented by the SNMP agent in Junos OS.

See the following sections for more information.

Junos OS Support for SNMP Versions

The Junos OS supports the following versions of SNMP. For more information, see Table 27 on page 372.

Table 27: Junos OS Support for SNMP Versions

SNMP Versions	Description
SNMPv1	The initial implementation of SNMP that defines the architecture and framework for SNMP.

Table 27: Junos OS Support for SNMP Versions *(Continued)*

SNMP Versions	Description
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	SNMPv3—The most up-to-date protocol focuses on security. SNMPv3 defines a security model, a 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

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.

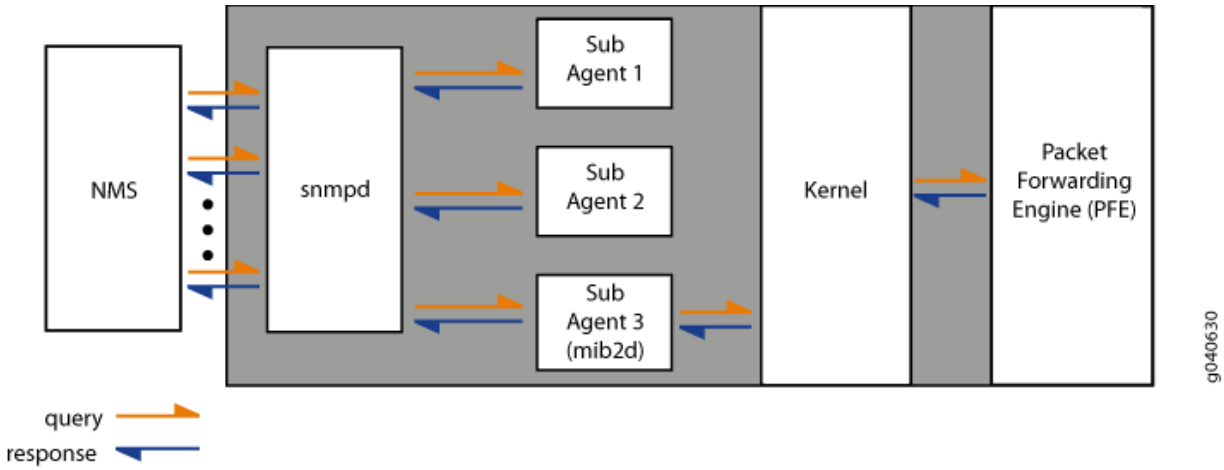
For more information about system logging severity levels for standard traps, see Standard SNMP Traps Supported by Junos OS . For more information about system logging severity levels for enterprise-specific traps, see Enterprise-Specific SNMP Traps Supported by Junos OS.

SNMP Communication Flow

When an NMS polls the primary agent for data, the primary agent immediately shares the data with the NMS if the requested data is available from the primary agent or one of the subagents. However, if the requested data does not belong to those categories that are maintained by the primary agent or the subagents, the subagent polls the Junos OS kernel or the process that maintains that data. On receiving the required data, the subagent passes the response back to the primary agent, which in turn passes it to the NMS.

Figure 21 on page 374 shows the communication flow among the NMS, SNMP primary agent (snmpd), SNMP subagents, Junos OS kernel, and the Packet Forwarding Engine.

Figure 21: SNMP Communication Flow



When a significant event, most often an error or a failure, occurs on a network device, the SNMP agent sends notifications to the SNMP manager. The SNMP implementation in Junos OS supports two types of notifications: traps and informs. *Traps* are unconfirmed notifications, whereas *informs* are confirmed notifications. Informs are supported only on devices that support SNMP version 3 (SNMPv3) configuration.

Trap Queuing

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

NOTE: You cannot configure trap queueing in Junos OS. You cannot view information about trap queues except for what is provided in the system logs.

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. Junos OS checks for the 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.

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, throttle queue and destination queue put together—is 40,000. However, on EX Series Ethernet Switches, the maximum size of the trap queue is 1,000. The maximum size of any one queue is 20,000 for devices other than EX Series Switches. On EX Series Switches, the maximum size of one queue is 500. When a trap is added to the throttle queue, or if the throttle queue has exceeded the maximum size, the trap is added back on 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.

Loading MIB Files to a Network Management System

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 Junos OS Enterprise MIBs index at https://www.juniper.net/documentation/en_US/release-independent/junos/mibs/mibs.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 SNMP MIB Explorer Download page for Juniper Networks SNMP MIB packages (SNMP MIB Explorer).
2. Click the **TAR** or **ZIP** link under the appropriate release heading to download the Junos MIB package for that release.
3. Decompress the file (**.tar** or **.zip**) using an appropriate utility.
4. 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-rfc4293.txt**
   g. **mib-rfc2012a.txt**
   h. **mib-rfc2013a.txt**
   i. **mib-rfc2571.txt**
   j. **mib-rfc2863a.txt**
   k. **mib-rfc4001.txt**
5. Load the remaining standard MIB files.

   NOTE: You must follow the order specified in this procedure. This is to ensure that you load standard MIBs before 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.

6. 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
   - **mib-jnx-cos.txt**
   - **mib-jnx-mimstp.txt**
   - **mib-jnx-l2cp-features.txt**
   - **mib-jnx-mpls-ldp.txt**
   - **mib-jnx-sp.txt**
   - **mib-jnx-ipforward.txt**
   - **mib-jnx-jsysmon.txt**
   - **mib-jnx-vpn.txt**
   - **mib-jnx-pwtdm.txt**
   - **mib-jnx-pwatm.txt**
   - **mib-jnx-mbg-smi.txt**
   - **mib-jnx-vpls-generic.txt**
   - **mib-jnx-vpls-ldp.txt**
   - **mib-jnx-vpls-bgp.txt**
   - **mib-jnx-mobile-gateways.txt**
   - **mib-jnx-optif.txt**
   - **mib-jnx-bl.txt**
   - **mib-jnx-gen-set.txt**
   - **mib-jnx-if-extensions.txt**

- **mib-jnx-if-accounting.txt**
- **mib-jnx-alarm.txt**
- **mib-jnx-dot3oam-capability.txt**
- **mib-jnx-ipmcast-capability.txt**

7. 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 are 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 are 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.

Understand the Integrated Local Management Interface

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 a 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`
- `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 `atmfMYIPNmAddress` or `atmfPortMyIfname` in the SNMP MIB Explorer.

SEE ALSO

Understanding Device Management Functions in Junos OS

Configure SNMP in Junos OS

IN THIS SECTION

- Configure SNMP | 379
- Configure SNMP Details | 388
- Configure the Commit Delay Timer | 391
- Configure SNMP on a Device Running Junos OS | 392
- Example: Configure SNMP on the QFabric System | 394

Configure SNMP

IN THIS SECTION

- Configuration Statements at the [edit snmp] Hierarchy Level | 380
- Configure Basic Settings for SNMP | 384

You can implement SNMP in the Junos OS Software running on the QFX Series and OCX Series products. By default, SNMP is not enabled. To enable SNMP, you must include the SNMP configuration statements at the `[edit]` hierarchy level.

To configure the minimum requirements for SNMP, include `community public` statement at the `[edit snmp]` hierarchy level.

To configure complete SNMP features, see *snmp*.

Configuration Statements at the [edit snmp] Hierarchy Level

This topic shows all 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 {
        alarm-management {
            alarm-list-name list-name {
                alarm-id id {
                    alarm-state state {
                        description alarm-description;
                        notification-id notification-id-of-alarm;
                        resource-prefix alarm-resource-prefix;
                        varbind-index varbind-index-in-alarm-varbind-list;
                        varbind-subtree alarm-varbind-subtree;
                        varbind-value alarm-varbind-value;
                    }
                }
            }
        }
    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;
        memory-trace;
        no-remote-trace;
        no-default-memory-trace;
    }
```

```
    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);
    }
}
```

Configure Basic Settings for SNMP

The following sections contain information about basic SNMP configuration and a few examples of configuring the basic SNMP operations on devices running Junos OS:

Configure Basic Settings for SNMPv1 and SNMPv2

You cannot enable SNMP on devices running Junos OS by default. To enable SNMP on devices running Junos OS, include the `community public` statement at the `[edit snmp]` hierarchy level.

Enabling SNMPv1 and SNMPv2 Get and GetNext Operations

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

A community that is defined as public grants access to all MIB data to any client.

To enable SNMPv1 and SNMPv2 `Set` operations on the device, you must include the following statements at the `[edit snmp]` hierarchy level:

Enabling SNMPv1 and SNMPv2 Set Operations

```
[edit snmp]
view all {
    oid .1;
}
community private {
    view all;
    authorization read-write;
}
```

The following example shows the basic minimum configuration for SNMPv1 and SNMPv2 traps on a device:

Configuring SNMPv1 and SNMPv2 Traps

```
[edit snmp]
trap-group jnpr {
    targets {
        192.168.69.179;
    }
}
```

Configure Basic Settings for SNMPv3

The following example shows the minimum SNMPv3 configuration for enabling `Get`, `GetNext`, and `Set` operations on a device (note that the configuration has authentication set to `md5` and privacy to `none`):

Enabling SNMPv3 Get, GetNext, and Set Operations

```
[edit snmp]
v3 {
    usm {
        local-engine {
            user jnpruser {
                authentication-md5 {
                    authentication-key "$9$guaDiQFnAuOQzevMWx7ikqP"; ## SECRET-DATA
                }
```

```
                privacy-none;
            }
        }
    }
    vacm {
        security-to-group {
            security-model usm {
                security-name jnpruser {
                    group grpnm;
                }
            }
        }
        access {
            group grpnm {
                default-context-prefix {
                    security-model any {
                        security-level authentication {
                            read-view all;
                            write-view all;
                        }
                    }
                }
            }
        }
    }
}
view all {
    oid .1;
}
```

The following example shows the basic configuration for SNMPv3 informs on a device (the configuration has authentication and privacy settings to `none`):

Configuring SNMPv3 Informs

```
[edit snmp]
v3 {
    usm {
        remote-engine 00000063200133a2c0a845c3 {
            user RU2_v3_sha_none {
                authentication-none;
                privacy-none;
```

```
        }
    }
}
vacm {
    security-to-group {
        security-model usm {
            security-name RU2_v3_sha_none {
                group g1_usm_auth;
            }
        }
    }
    access {
        group g1_usm_auth {
            default-context-prefix {
                security-model usm {
                    security-level authentication {
                        read-view all;
                        write-view all;
                        notify-view all;
                    }
                }
            }
        }
    }
}
target-address TA2_v3_sha_none {
    address 192.168.69.179;
    tag-list tl1;
    address-mask 255.255.252.0;
    target-parameters TP2_v3_sha_none;
}
target-parameters TP2_v3_sha_none {
    parameters {
        message-processing-model v3;
        security-model usm;
        security-level none;
        security-name RU2_v3_sha_none;
    }
    notify-filter nf1;
}
notify N1_all_tl1_informs {
    type inform; # Replace inform with trap to convert informs to traps.
    tag tl1;
```

```
    }
    notify-filter nf1 {
        oid .1 include;
    }
}
view all {
    oid .1 include;
}
```

You can convert the SNMPv3 informs to traps by setting the value of the `type` statement at the `[edit snmp v3 notify N1_all_tl1_informs]` hierarchy level to `trap` as shown in the following example:

Converting Informs to Traps

```
user@host# set snmp v3 notify N1_all_tl1_informs type trap
```

SEE ALSO

- Understand SNMP Implementation in Junos OS | 371
- *snmp*
- Monitor SNMP Activity and Track Problems That Affect SNMP Performance on a Device Running Junos OS | 514
- Optimize the Network Management System Configuration for the Best Results | 405
- Configure Options on Managed Devices for Better SNMP Response Time | 399
- No Link Title

Configure SNMP Details

You can use SNMP to store basic administrative details, such as a contact name and the location of the device. Your management system can then retrieve this information remotely when you are troubleshooting an issue or performing an audit. In SNMP terminology, these are the sysName, sysContact, sysDescription, and sysLocation objects found within the system group of MIB-2 (as defined in RFC 1213, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II*). You can set initial values directly in the Junos OS configuration for each system being managed by SNMP.

NOTE: For the devices that are managed by SNMP, always keep the name, location, contact, and description information configured and updated.

To set the SNMP details:

1. Configure a system name.

   Set the system name details by including the `name` statement at the `[edit snmp]` hierarchy level.

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

   For example:

   ```
   [edit groups global snmp]
   user@host# set name "host” # Overrides the system name
   ```

2. Configure a system contact.

   Set the system contact details by including the `contact` statement at the `[edit snmp]` hierarchy level, or in an appropriate configuration group as shown here.

   This administrative contact is placed into the MIB II sysContact object.

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

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

   For example:

   ```
   [edit groups global snmp]
   user@host# set contact "Enterprise Support, (650) 555-1234" # Specifies the name and phone
   number of the
   administrator.
   ```

3. Configure a system description.

This string is placed into the MIB II sysDescription object. If the description contains spaces, enclose it in quotation marks (" ").

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

For example:

```
[edit groups global snmp]
user@host# set description "M10i router with 8 FPCs" # Specifies the description for the
device.
```

4. Configure a system location.

   This string is placed into the MIB II sysLocation object. If the location contains spaces, enclose it in quotation marks (" ").

   To specify the system location:

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

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

For example:

```
[edit groups global snmp]
user@host# set location "London Corporate Office, Lab 5, Row 11, Rack C" # Specifies the
location of the device.
```

5. At the top level of the configuration, apply the configuration group.

   If you use a configuration group, you must apply it for it to take effect.

```
[edit]
user@host# set apply-groups global
```

6. Commit the configuration.

```
user@host# commit
```

7. To verify the configuration, enter the `show snmp mib walk system` operational-mode command.

   The `show snmp mib walk system` command performs a MIB walk through of the system table (from MIB-2 as defined in RFC 1213). The SNMP agent in Junos OS responds by printing each row in the table and its associated value. You can use the same command to perform a MIB walk through any part of the MIB tree supported by the agent.

```
user@host> show snmp mib walk system
sysDescr.0    = M10i router with 8 FPCs
sysObjectID.0 = jnxProductNameM10i
sysUpTime.0   = 173676474
sysContact.0  = Enterprise Support, (650) 555-1234
sysName.0     = host
sysLocation.0 = London Corporate Office, Lab 5, Row 11, Rack C
sysServices.0 = 4
```

Configure the Commit Delay Timer

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 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 *Junos OS CLI User Guide* .

Configure SNMP on a Device Running Junos OS

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 `community public` statement at the `[edit snmp]` hierarchy level.

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);
    }
}
```

SEE ALSO

Understand SNMP Implementation in Junos OS | 371

Example: Configure SNMP on the QFabric System

IN THIS SECTION

- Requirements | 394
- Overview | 395
- Configuration | 395

By default, SNMP is disabled on devices running Junos OS. This example describes the steps for configuring SNMP on the QFabric system.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 12.2
- Network management system (NMS) (running the SNMP manager)
- QFabric system (running the SNMP agent) with multiple Node devices

Overview

IN THIS SECTION

- Topology | 395

You must enable SNMP on your device by including configuration statements at the `[edit snmp]` hierarchy level. At a minimum, you must configure the `community public` statement. The community defined as public grants read-only access to MIB data to any client.

If no `clients` statement is configured, all clients are allowed. We recommend that you always include the `restrict` option to limit SNMP client access to the switch.

Topology

The network topology in this example includes an NMS, a QFabric system with four Node devices, and external SNMP servers that are configured for receiving traps.

Configuration

IN THIS SECTION

- Procedure | 395

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `[edit]` hierarchy level.

```
set snmp name “snmp qfabric” description “qfabric0 switch”
set snmp location “Lab 4 Row 11” contact “qfabric-admin@qfabric0”
set snmp community public authorization read-only
set snmp client-list list0 192.168.0.0/24
```

```
set snmp community public client-list-name list0
set snmp community public clients 192.170.0.0/24 restrict
set snmp trap-group “qf-traps” destination-port 155 targets 192.168.0.100
```

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure SNMP on the QFabric system:

NOTE: If the name, description, location, contact, or community name contains spaces, enclose the text in quotation marks (" ").

1. Configure the SNMP system name:

```
[edit snmp]
user@switch# set name “snmp qfabric”
```

NOTE: You can access the above configured SNMP system name:

- By doing a query with the SNMPGet on policy object identifier (OID) sysName.0.
- From the generic jnxSyslogTrap. To send the jnxSyslogTrap, configure the trap events at `[edit event-options policy]` hierarchy.

2. Specify a description.

```
[edit snmp]
user@switch# set description “qfabric0 system”
```

This string is placed into the MIB II sysDescription object.

3. Specify the physical location of the QFabric system.

```
[edit snmp]
user@switch# set location “Lab 4 Row 11”
```

This string is placed into the MIB II sysLocation object.

4. Specify an administrative contact for the SNMP system.

```
[edit snmp]
user@switch# set contact “qfabric-admin@qfabric0”
```

This name is placed into the MIB II sysContact object.

5. Specify a unique SNMP community name and the read-only authorization level.

NOTE: The `read-write` option is not supported on the QFabric system.

```
[edit snmp]
user@switch# set community public authorization read-only
```

6. Create a client list with a set of IP addresses that can use the SNMP community.

```
[edit snmp]
user@switch# set client-list list0 192.168.0.0/24
user@switch# set community public client-list-name list0
```

7. Specify IP addresses of clients that are restricted from using the community.

```
[edit snmp]
user@switch# set community public clients 198.51.100.0/24 restrict
```

8. Configure a trap group, destination port, and a target to receive the SNMP traps in the trap group.

```
[edit snmp]
user@switch# set trap-group “qf-traps” destination-port 155 targets 192.168.0.100
```

NOTE: You do not need to include the `destination-port` statement if you use the default port 162.

The trap group qf-traps is configured to send traps to 192.168.0.100.

Results

From 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]
user@switch# show
    snmp {
        name "snmp qfabric";
        description "qfabric0 system";
        location "Lab 4 Row 11";
        contact "qfabric-admin@qfabric0";
        client-list list0 {
            192.168.0.0/24;
        }
        community public {
            authorization read-only;
            clients {
                198.51.100.0/24 restrict;
            }
        }
        trap-group qf-traps {
            destination-port 155;
            targets {
                192.168.0.100;
            }
        }
    }
```

If you are done configuring the device, enter `commit` from configuration mode.

SEE ALSO

Understanding the Implementation of SNMP on the QFabric System

snmp

Configure Options on Managed Devices for Better SNMP Response Time

IN THIS SECTION

- Enable the stats-cache-lifetime Option | 399
- Filter Out Duplicate SNMP Requests | 400
- Exclude Interfaces That Are Slow in Responding to SNMP Queries | 400

The following sections contain information about configuration options on the managed devices that can enhance SNMP performance:

Enable the stats-cache-lifetime Option

Junos OS provides you with an option to configure the length of time (in seconds) the interface stats are cached. If the NMS queries again for the same interface within the cache time, the same data is returned. If the NMS queries after the cache time, the cache is no longer valid, fresh data is fetched from the lower layers, and the cache timestamp is updated. The default `stats-cache-lifetime` is 5 seconds. This can be tuned as per the polling frequency.

NOTE: Reducing the value of the stats-cache-lifetime option results in more queries and can impact performance. To get the live uncached statistics, set the value of the stats-cache-lifetime option to 0. However, this is not recommended since it completely disables the caching feature and impacts performance.

Filter Out Duplicate SNMP Requests

If a network management station retransmits a `Get`, `GetNext`, or `GetBulk` SNMP request too frequently to a device, 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. The Junos OS enables you to filter out duplicate `Get`, `GetNext`, and `GetBulk` SNMP requests. The 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

NOTE: By default, filtering of duplicate SNMP requests is disabled on devices running the Junos OS.

To enable filtering of duplicate SNMP requests on devices running the Junos OS, include the `filter-duplicates` statement at the `[edit snmp]` hierarchy level:

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

Exclude Interfaces That Are Slow in Responding to SNMP Queries

An interface that is slow in responding to SNMP requests for interface statistics can delay the kernel responses to SNMP requests. You can review the mib2d log file to find out how long the kernel takes to respond to various SNMP requests. For more information about reviewing the log file for the kernel response data, see “Checking Kernel and Packet Forwarding Engine Response” under "Monitoring SNMP Activity and Tracking Problems That Affect SNMP Performance on a Device Running Junos OS" on page 514.

If you notice that a particular interface is slow in responding and think that it is slowing down the kernel from responding to SNMP requests, exclude that interface from the SNMP queries to the device. You can exclude an interface from the SNMP queries either by configuring the `filter-interface` statement or by modifying the SNMP view settings.

The following example shows a sample configuration for excluding interfaces from the SNMP `Get`, `GetNext`, and `Set` operations:

```
[edit]
    snmp {
    filter-interfaces {
        interfaces { # exclude the specified interfaces
            interface1;
            interface2;
        }
        all-internal-interfaces; # exclude all internal interfaces
    }
}
```

The following example shows the SNMP view configuration for excluding the interface with an interface index (ifIndex) value of 312 from a request for information related to the ifTable and ifXtable objects:

```
[edit snmp]
    view test {
        oid .1 include;
        oid ifTable.1.*.312 exclude;
        oid ifXTable.1.*.312 exclude
    }
```

Alternatively, you can take the interface that is slow in responding offline.

RELATED DOCUMENTATION

Understand SNMP Implementation in Junos OS | **371**

Monitor SNMP Activity and Track Problems That Affect SNMP Performance on a Device Running Junos OS | **514**

No Link Title

No Link Title

Enterprise Specific Utility MIB to Enhance SNMP Coverage

IN THIS SECTION

- Utility MIB | 402

Utility MIB

IN THIS SECTION

- Use the Enterprise-Specific Utility MIB to Enhance SNMP Coverage | 403

The Juniper Networks enterprise-specific Utility MIB, whose object ID is {jnxUtilMibRoot 1}, defines objects for counters, integers, and strings. The Utility MIB contains one table for each of the following five data types:

- 32-bit counters
- 64-bit counters
- Signed integers
- Unsigned integers
- Octet strings

You can use these containers MIB objects to store the data that are not supported for SNMP operations. You can populate data for these objects either by using CLI commands or with the help of Op scripts and an RPC API that can invoke the CLI commands.

Each data type has an arbitrary ASCII name, which is defined when the data is populated, and a timestamp that shows the last time when the data instance was modified. For a downloadable version of this MIB, see Routing Policies, Firewall Filters, and Traffic Policers User Guide.

For information about the enterprise-specific Utility MIB objects, see the following topics:

- jnxUtilCounter32Table
- jnxUtilCounter64Table
- jnxUtilIntegerTable
- jnxUtilUintTable
- jnxUtilStringTable

Use the Enterprise-Specific Utility MIB to Enhance SNMP Coverage

You might need to have customized performance metrics even though the Junos OS has built-in performance metrics and monitoring options. To make it easier for you to monitor such customized data through a standard monitoring system, the Junos OS provides you with an enterprise-specific Utility MIB that can store such data and thus extend SNMP support for managing and monitoring the data of your choice.

The following CLI commands enable you to set and clear Utility MIB object values:

- `request snmp utility-mib set instance` *`name`* `object-type <counter | counter 64 | integer | string | unsigned integer> object-value` *`value`*
- `request snmp utility-mib clear instance` *`name`* `object-type <counter | counter 64 | integer | string | unsigned integer>`

The `instance` *`name`* option of the `request snmp utility-mib <set | clear>` command specifies the name of the data instance and is the main identifier of the data. The `object-type <counter | counter 64 | integer | string | unsigned integer>` option enables you to specify the object type, and the `object-value` *`value`* option enables you to set the value of the object.

To automate the process of populating Utility MIB data, you can use a combination of an event policy and event script. The following examples show the configuration for an event policy to run `show system buffers` every hour and to store the `show system buffers` data in Utility MIB objects by running an event script (`check-mbufs.slax`).

Event Policy Configuration

To configure an event policy that runs the `show system buffers` command every hour and invokes `check-mbufs.slax` to store the `show system buffers` data into Utility MIB objects, include the following statements at the [`edit`] hierarchy level:

```
event-options {
    generate-event {
        1-HOUR time-interval 3600;
    }
    policy MBUFS {
        events 1-HOUR;
        then {
            event-script check-mbufs.slax; # script stored at /var/db/scripts/event/
        }
    }
    event-script {
        file check-mbufs.slax;
    }
}
```

check-mbufs.slax Script

The following example shows the `check-mbufs.slax` script that is stored under `/var/db/scripts/event/`:

```
------ script START ------
version 1.0;

ns junos = "http://xml.juniper.net/junos/*/junos";
ns xnm = "http://xml.juniper.net/xnm/1.1/xnm";
ns jcs = "http://xml.juniper.net/junos/commit-scripts/1.0";
ns ext = "http://xmlsoft.org/XSLT/namespace";

match / {
    <op-script-results>{
        var $result = jcs:invoke("get-buffer-informations");
        var $rpc = <request-snmp-utility-mib-set> {
                <object-type> "integer";
                <instance> "current-mbufs";
                <object-value> $result/current-mbufs;
        }
```

```
        var $res = jcs:invoke($rpc);
        expr jcs:syslog("external.info", $res/..//snmp-utility-mib-results/snmp-utility-mib-
result);
    }
}
------ script END ------
```

You can run the following command to check the data stored in the Utility MIB as a result of the event policy and script shown in the preceding examples:

```
user@host> show snmp mib walk jnxUtilData ascii jnxUtilIntegerValue."current-mbufs" = 0
jnxUtilIntegerTime."current-mbufs" = 07 da 05 0c  03 14 2c 00  2d 07 00   user@caramels>
```

NOTE: The `show snmp mib walk` command is not available on the QFabric system, but you can use external SNMP client applications to perform this operation.

SEE ALSO

- Enterprise-Specific SNMP MIBs Supported by Junos OS
- Standard SNMP MIBs Supported by Junos OS
- *Understanding the Implementation of SNMP on the QFabric System*

Optimize the Network Management System Configuration for the Best Results

You can modify your network management system configuration to optimize the response time for SNMP queries. You can configure the network management system by following the below tips:

- **Change the Polling Method from Column-by-Column to Row-by-Row**

  You can configure the network management system to use the row-by-row method for SNMP data polling. It is evident that row-by-row and multiple row-by-multiple-row polling methods are more efficient than column-by-column polling.

By configuring the network management system to use the row-by-row data polling method, you can:

- Poll the data for only one interface in a request instead of a single request polling data for multiple interfaces as in the case with column-by-column polling.
- Reduces the risk of requests timing out.

- **Reduce the Number of Variable Bindings per PDU**

  You can improve the response time for SNMP requests by reducing the number of variable bindings per protocol data unit (PDU). A request that polls for data related to multiple objects mapped to different index entries, translate into multiple requests at the device end. This is because the subagent might have to poll different modules to obtain data linked to different index entries.

  The recommended method is to ensure that a request has only objects linked to one index entry instead of multiple objects linked to different index entries.

  > **NOTE**: If responses from a device are slow, avoid using the `GetBulk` option for the device, because a `GetBulk` request might contain objects that are linked to various index entries and might further increase the response time.

- **snmp bulk-get recommended number of OIDs and max-repetitions**

  An SNMP bulk-get request responds with a total of (max-repetitions * number-of-OIDs) variable bindings. When interface statistics objects (such as ifInOctets, ifOutOctets, etc) are present in a query, the requests are sent to lower layers. Hence, there is an impact on the responses by an increase in the max-repetitions that you send in a bulk-get request. For bulk-get queries for interface stats objects, it is recommended to use the 'max-repetitions' value of 10, and the maximum number of OIDs per request is 10.

- **Increase Timeout Values in Polling and Discovery Intervals**

  By increasing the timeout values for polling and discovery intervals, you can:

  - Increase the queuing time at the device end.
  - Reduce the number of throttle drops that occur because of the request timing out.

- **Reduce Incoming Packet Rate at the snmpd**

  The following methods reduce the risk of SNMP requests piling up on any device.

  - Reduce the frequency of sending SNMP requests to a device.
  - Increase the polling interval.

- Control the use of `GetNext` requests.
- Reduce the number of polling stations per device.

RELATED DOCUMENTATION

Understanding SNMP Implementation in Junos OS

Monitor SNMP Activity and Track Problems That Affect SNMP Performance on a Device Running Junos OS | **514**

Managing Traps and Informs

Interfaces to Accept SNMP Requests

IN THIS SECTION

- Configure the Interfaces on Which SNMP Requests Can Be Accepted | **407**
- Configure a Proxy SNMP Agent | **407**
- Example: Configure Secured Access List Checking | **408**
- Filter Interface Information Out of SNMP Get and GetNext Output | **409**

Configure the Interfaces on Which SNMP Requests Can Be Accepted

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.

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.

Configure a Proxy SNMP Agent

Starting with Release 12.3, Junos OS enables you to assign one of the devices in the network as a proxy SNMP agent through which the network management system (NMS) can query other devices in the

network. When you configure a proxy, you can specify the names of devices to be managed through the proxy SNMP agent.

When the NMS queries the proxy SNMP agent, the NMS specifies the community name (for SNMPv1 and SNMPv2) or the context and security name (for SNMPv3) associated with the device from which it requires the information.

NOTE: If you have configured authentication and privacy methods and passwords for SNMPv3, those parameters are also specified in the query for SNMPv3 information.

To configure a proxy SNMP agent and specify devices to be managed by the proxy SNMP agent, see *proxy (snmp)*.

NOTE: Starting with Junos OS Release 15.2, you must configure the `interface <interface-name>` statement at the `[edit snmp]` hierarchy level for the proxy SNMP agent.

NOTE: The community and security configurations for the proxy should match the corresponding configurations on the device that is to be managed.

NOTE: The devices managed by the proxy SNMP agent send the traps directly to the network management system since the proxy SNMP agent does not have trap-forwarding capabilities.

You can use the `show snmp proxy` operational mode command to view proxy details on a device. The `show snmp proxy` command returns the proxy names, device names, SNMP version, community/security, and context information.

Example: Configure Secured Access List Checking

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 ];
}
```

Filter Interface Information Out of SNMP Get and GetNext Output

Junos OS enables you to filter out information related to specific interfaces from the output of SNMP `Get` and `GetNext` requests. You can perform this 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 {
            interface-name 1;
            interface-name 2;
        }
        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 only applicable to SNMP operations. 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.

Configure SNMP for Routing Instances

IN THIS SECTION

- Understand SNMP Support for Routing Instances | **411**
- SNMPv3 Management Routing Instance | **412**
- SNMP MIBs Supported for Routing Instances | **413**
- Support Classes for MIB Objects | **425**
- SNMP Traps Supported for Routing Instances | **426**
- Identify a Routing Instance | **427**
- Enable SNMP Access over Routing Instances | **428**
- Specify a Routing Instance in an SNMPv1 or SNMPv2c Community | **428**
- Example: Configure Interface Settings for a Routing Instance | **429**
- Configure Access Lists for SNMP Access over Routing Instances | **431**

Understand SNMP Support for Routing Instances

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.

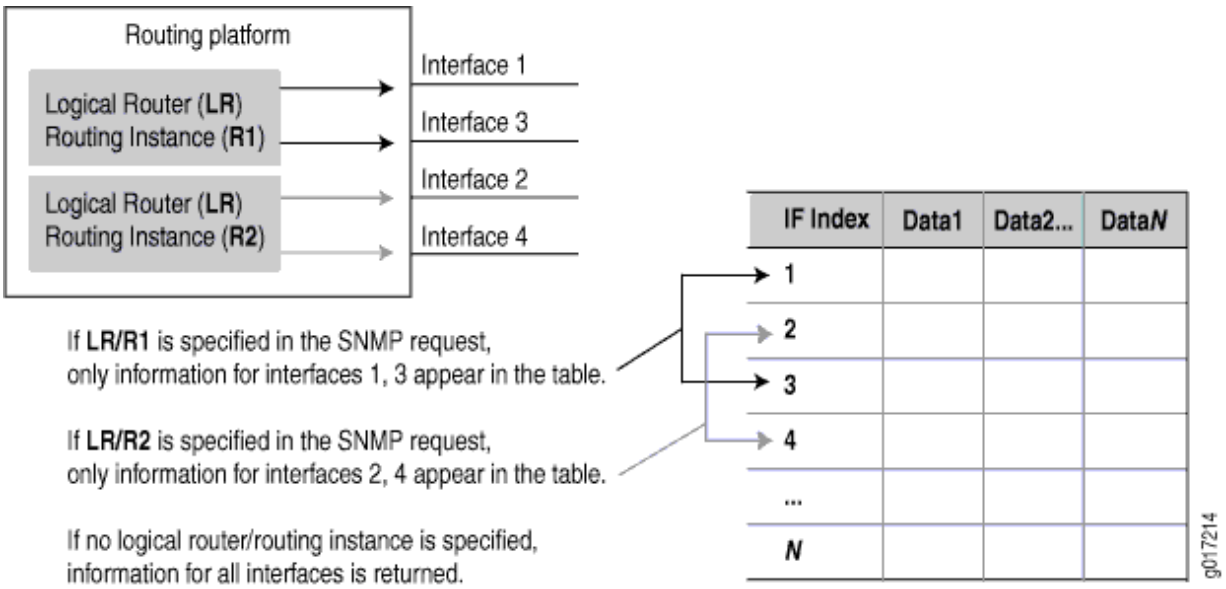
In Junos OS:

- Clients from routing instances and/or logical systems other than the default can access MIB objects and perform SNMP operations only on the routing instance and/or 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.
- The Junos management routing instance (`mgmt_junos`) is a special instance. Clients from the management routing instance are treated as if they were in the default routing instance and 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 22 on page 411). Service providers can use this information for their own management needs or export the data for use by their customers.

Figure 22: 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.

SNMPv3 Management Routing Instance

IN THIS SECTION

- Benefits | 412
- Enable the Management Routing Instance | 412
- Remove the Management Routing Instance | 413

Starting in Junos OS 19.4R1, you can access information related to all routing instances and logical system networks and not specific to ingress routing instance by configuring the SNMPv3 management interface in a required routing instance. You can configure the management instance configuration statement at the `[edit snmp v3]` hierarchy level.

Benefits

SNMPv3 management routing instance enables all the SNMPv3 requests from non-default routing instance as if the requests are from default routing instance. Using SNMPv3 management routing instance, you access the information related to all routing instances and logical system networks.

Enable the Management Routing Instance

To enable the SNMPv3 management routing instance:

1. Configure the management-instance statement.

```
[edit]
user@host# set snmp v3 management-routing-instance <routing-instance>
```

2. Commit the configuration.

```
[edit]
user@host# commit
```

Remove the Management Routing Instance

To remove the SNMPv3 management routing instance:

1. Delete or deactivate the SNMPv3 management routing instance statement.

```
[edit]
user@host# delete snmp v3 management-routing-instance <routing-instance>
```

You cannot configure the Junos management routing instance (`mgmt_junos`) at the [`edit snmp v3 management-routing-instance <routing-instance>`] hierarchy level since the `mgmt_junos` has the access to all routing instances by default.

SNMP MIBs Supported for Routing Instances

Table 28 on page 413 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 28: MIB Support for Routing Instances (Juniper Networks MIBs)

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

Table 28: MIB Support for Routing Instances (Juniper Networks MIBs) *(Continued)*

Object	Support Class	Description/Notes
jnxServices(2)	–	Services
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.

Table 28: MIB Support for Routing Instances (Juniper Networks MIBs) *(Continued)*

Object	Support Class	Description/Notes
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.
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.

Table 28: MIB Support for Routing Instances (Juniper Networks MIBs) *(Continued)*

Object	Support Class	Description/Notes
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.
jnxVpnMIB(26)	Class 2	All instances within a logical system are exposed. Data will not be segregated down to the routing instance level.

Table 28: MIB Support for Routing Instances (Juniper Networks MIBs) *(Continued)*

Object	Support Class	Description/Notes
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 29 on page 417 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 29: 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

Table 29: Class 1 MIB Objects (Standard and Juniper MIBs) *(Continued)*

Class	MIB	Objects
	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 29: 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
Class 1	rfc2863a.mib	Examples: ifXtable ifStackTable
	rfc2665a.mib	etherMIB
	rfc2515a.mib	atmMIB objects Examples: atmInterfaceConfTable atmVplTable atmVclTable

Table 29: Class 1 MIB Objects (Standard and Juniper MIBs) *(Continued)*

Class	MIB	Objects
	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

Table 29: Class 1 MIB Objects (Standard and Juniper MIBs) *(Continued)*

Class	MIB	Objects
	jnx-atm.mib	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

Table 29: Class 1 MIB Objects (Standard and Juniper MIBs) *(Continued)*

Class	MIB	Objects
	jnx-services.mib	Example: jnxSvcFlowTableAggStatsTable
	jnx-coll.mib	jnxCollectorMIB Examples: jnxCollPicIfTable jnxCollFileEntry

Table 30 on page 422 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 30: 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 **NOTE**: The `igmpmib.mib` is the draft version of the IGMP Standard MIB in the experimental tree. Junos OS does not support the original IGMP Standard MIB.

Table 30: Class 2 MIB Objects (Standard and Juniper MIBs) *(Continued)*

Class	MIB	Objects
	l3vpnmib.mib	mplsVpnmib
	jnx-mpls.mib	Example: mplsLspList
	jnx-ldp.mib	jnxLdp Example: jnxLdpStatsTable
	jnx-vpn.mib	jnxVpnMIB
	jnx-bgpmib2.mib	jnxBgpM2Experiment

Table 31 on page 423 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 31: 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

Table 31: Class 3 MIB Objects (Standard and Juniper MIBs) *(Continued)*

Class	MIB	Objects
	jnx-chassis-alarm.mib	jnxAlarms By default, SRX Series Firewalls queries jnxAlarms mib only on the primary node of redundancy group 0 (RG0) and not on the secondary node.
	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 32 on page 425 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 32: 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

Support Classes for MIB Objects

When a routing instance is specified, all routing-related MIB objects return data maintained by the routing instance in the request. For all other MIB objects, the data returned is segregated according to that routing instance. For example, only those interfaces assigned to that routing instance (for example, the logical interfaces [ifls] as well as their corresponding physical interfaces [ifds]) are exposed by the SNMP agent. Similarly, objects with an unambiguous attachment to an interface (for example, addresses) are segregated as well.

For those objects where the attachment is ambiguous (for example, objects in sysApplMIB), no segregation is done and all instances are visible in all cases.

Another category of objects is visible only when no logical system is specified (only within the default logical system) regardless of the routing instance within the default logical system. Objects in this category are Chassis MIB objects, objects in the SNMP group, RMON alarm, event and log groups, Ping MIB objects, configuration management objects, and V3 objects.

In summary, to support routing instances, MIB objects fall into one of the following categories:

- Class 1—Data is segregated according to the routing instance in the request. This is the most granular of the segregation classes.
- Class 2—Data is segregated according to the logical system specified in the request. The same data is returned for all routing instances that belong to a particular logical system. Typically, this applies to routing table objects where it is difficult to extract routing instance information or where routing instances do not apply.
- Class 3—Data is exposed only for the default logical system. The same set of data is returned for all routing instances that belong to the default logical system. If you specify another logical system (not the default), no data is returned. Typically this class applies to objects implemented in subagents that do not monitor logical system changes and register their objects using only the default context (for example, Chassis MIB objects).
- Class 4—Data is not segregated by routing instance. The same data is returned for all routing instances. Typically, this applies to objects implemented in subagents that monitor logical system changes and register or deregister all their objects for each logical system change. Objects whose values cannot be segregated by routing instance fall into this class.

See "SNMP MIBs Supported for Routing Instances" on page 413 for a list of the objects associated with each class.

SNMP Traps Supported for Routing Instances

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).

Identify a Routing Instance

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 5G Universal Routing Platforms), 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.

Enable SNMP Access over Routing Instances

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.

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. This setting applies to requests for any version of SNMP (SNMP v1, v2, or v3).

Specify a Routing Instance in an SNMPv1 or SNMPv2c Community

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;
            }
        }
    }
}
```

Example: Configure Interface Settings for a Routing Instance

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
```

Configure Access Lists for SNMP Access over Routing Instances

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, and applies to requests for any version of SNMP.

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.

Configure SNMP Remote Operations

IN THIS SECTION

- SNMP Remote Operations Overview | **432**
- Use the Ping MIB for Remote Monitoring Devices Running Junos OS | **436**
- Start a Ping Test | **437**
- Monitor a Running Ping Test | **438**
- Gather Ping Test Results | **442**
- Stop a Ping Test | **444**
- Interpret Ping Variables | **444**
- Use the Traceroute MIB for Remote Monitoring Devices Running Junos OS | **445**
- Start a Traceroute Test | **445**
- Monitor a Running Traceroute Test | **446**
- Monitor Traceroute Test Completion | **450**
- Gather Traceroute Test Results | **451**
- Stop a Traceroute Test | **453**
- Interpret Traceroute Variables | **453**

SNMP Remote Operations Overview

IN THIS SECTION

- SNMP Remote Operation Requirements | **433**

- Set SNMP Views | **433**
- Set Trap Notification for Remote Operations | **435**
- Use Variable-Length String Indexes | **435**
- Enable Logging | **436**

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.

Before starting the Ping MIB, see "Starting a Ping Test" on page 437.

Before starting the Traceroute MIB, see "Starting a Traceroute Test" on page 445.

Set 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: Set 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 Configure SNMP Communities and *community (SNMP)*.

For more information about the `view` statement, see Configure MIB Views, *view (SNMP Community)*, and *view (Configuring a MIB View)*.

Set 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: Set 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 Configure SNMP Trap Groups.

Use 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.

Enable 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;
    }
}
```

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).

Use the Ping MIB for Remote Monitoring Devices Running Junos OS

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 Ping MIB.

Start a Ping Test

IN THIS SECTION

- Before You Begin | 437
- Start a Ping Test | 437
- Use Multiple Set PDUs | 438
- Use a Single Set PDU | 438

Use this topic to launch an ICMP ping test. There are two ways to start a ping test: using multiple Set protocol data units (PDUs) or using a single Set PDU.

Before You Begin

Before you start a ping test, configure a Ping MIB view. This allows SNMP `Set` requests on `pingMIB`. For more information, see Configure MIB Views.

Starting in Junos OS Release 17.2X75-D100, you must configure RPM before starting an ICMP ping. Configure RPM using the `edit services rpm` command.

Start a Ping Test

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 446.

Read the following sections for how to order the variables.

Use Multiple Set 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`

Use 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`

Monitor a Running Ping Test

IN THIS SECTION

- pingResultsTable | 439
- pingProbeHistoryTable | 440
- Generate Traps | 441

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

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 440.

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.

Generate 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* and "Example: Setting Trap Notification for Remote Operations" on page 446.

Gather Ping Test Results

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 446. For more information about Ping MIB traps, see "Generating Traps" on page 446.

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 33 on page 442.

Table 33: 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 34 on page 443.

Table 34: 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
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 35 on page 443.

Table 35: 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.

Stop a Ping Test

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.

Interpret Ping Variables

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.

Use the Traceroute MIB for Remote Monitoring Devices Running Junos OS

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.

Start a Traceroute Test

IN THIS SECTION

- Use Multiple Set PDUs | 446
- Use a Single Set PDU | 446

Before you start a traceroute test, configure a Traceroute MIB view. This allows SNMP `Set` requests on `traceroutMIB`. To start a test, create a row in `traceRouteCtlTable` and set `traceRouteCtlAdminStatus` to `enabled`. You must specify at least the following before setting `traceRouteCtlAdminStatus` to `enabled`:

- `traceRouteCtlOwnerIndexSnmpAdminString`
- `traceRouteCtlTestNameSnmpAdminString`
- `traceRouteCtlTargetAddressInetAddress`
- `traceRouteCtlRowStatusRowStatus`

For all other values, defaults are chosen unless otherwise specified. `traceRouteCtlOwnerIndex` and `traceRouteCtlTestName` are used as the index, so their values are specified as part of the OID. To create a row, set `traceRouteCtlRowStatus` to `createAndWait` or `createAndGo` on a row that does not already exist. A value of `active` for `traceRouteCtlRowStatus` indicates that all necessary information has been specified and the test can begin; `traceRouteCtlAdminStatus` can be set to `enabled`. An SNMP `Set` request that sets `traceRouteCtlRowStatus` 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 446.

There are two ways to start a traceroute test:

Use Multiple Set 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:

- `traceRouteCtlRowStatus` to createAndWait
- All appropriate test variables
- `traceRouteCtlRowStatus` to `active`

  The Junos OS now verifies that all necessary information to run a test has been specified.
- `traceRouteCtlAdminStatus` to `enabled`

Use 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:

- `traceRouteCtlRowStatus` to `createAndGo`
- All appropriate test variables
- `traceRouteCtlAdminStatus` to enabled

Monitor a Running Traceroute Test

IN THIS SECTION

- traceRouteResultsTable | 447
- traceRouteProbeResultsTable | 448
- traceRouteHopsTable | 449
- Generate Traps | 450

When traceRouteCtlAdminStatus is successfully set to enabled, the following is done before the acknowledgment of the SNMP Set request is sent back to the client:

- traceRouteResultsEntry is created if it does not already exist.
- traceRouteResultsOperStatus transitions to enabled.

For more information, see the following sections:

traceRouteResultsTable

While the test is running, this traceRouteResultsTable keeps track of the status of the test. The value of traceRouteResultsOperStatus is enabled while the test is running and disabled when it has stopped.

The value of traceRouteCtlAdminStatus remains enabled until you set it to disabled. Thus, to get the status of the test, you must examine traceRouteResultsOperStatus.

The traceRouteCtlFrequency variable can be used to schedule many tests for one traceRouteCtlEntry. After a test ends normally (you did not stop the test) and traceRouteCtlFrequency number of seconds has elapsed, the test is started again just as if you had set traceRouteCtlAdminStatus to enabled. If you intervene at any time between repeated tests (you set traceRouteCtlAdminStatus to disabled or traceRouteCtlRowStatus to notInService), the repeat feature is disabled until another test is started and ends normally. A value of 0 for traceRouteCtlFrequency indicates this repeat feature is not active.

traceRouteResultsIpTgtAddr and traceRouteResultsIpTgtAddrType are set to the value of the resolved destination address when the value of traceRouteCtlTargetAddressType is dns. When a test starts successfully and traceRouteResultsOperStatus transitions to enabled:

- traceRouteResultsIpTgtAddr is set to null-string.
- traceRouteResultsIpTgtAddrType is set to unknown.

traceRouteResultsIpTgtAddr and traceRouteResultsIpTgtAddrType are not set until traceRouteCtlTargetAddress can be resolved to a numeric address. To retrieve these values, poll traceRouteResultsIpTgtAddrType for any value other than unknown after successfully setting traceRouteCtlAdminStatus to enabled.

At the start of a test, traceRouteResultsCurHopCount is initialized to traceRouteCtlInitialTtl, and traceRouteResultsCurProbeCount is initialized to 1. Each time a probe result is determined, traceRouteResultsCurProbeCount increases by 1. While the test is running, the value of traceRouteResultsCurProbeCount reflects the current outstanding probe for which results have not yet been determined.

The traceRouteCtlProbesPerHop number of probes is sent for each time-to-live (TTL) value. When the result of the last probe for the current hop is determined, provided that the current hop is not the destination hop, traceRouteResultsCurHopCount increases by 1, and traceRouteResultsCurProbeCount resets to 1.

At the start of a test, if this is the first time this test has been run for this traceRouteCtlEntry, traceRouteResultsTestAttempts and traceRouteResultsTestSuccesses are initialized to 0.

At the end of each test execution, traceRouteResultsOperStatus transitions to disabled, and traceRouteResultsTestAttempts increases by 1. If the test was successful in determining the full path to

the target, traceRouteResultsTestSuccesses increases by 1, and traceRouteResultsLastGoodPath is set to the current time.

traceRouteProbeResultsTable

Each entry in traceRouteProbeHistoryTable is indexed by five variables:

- The first two variables, traceRouteCtlOwnerIndex and traceRouteCtlTestName, are the same ones used for traceRouteCtlTable and to identify the test.
- The third variable, traceRouteProbeHistoryIndex, is a counter, starting from 1 and wrapping at FFFFFFFF. The maximum number of entries is limited by traceRouteCtlMaxRows.
- The fourth variable, traceRouteProbeHistoryHopIndex, indicates which hop this probe is for (the actual time-to-live or TTL value). Thus, the first traceRouteCtlProbesPerHop number of entries created when a test starts have a value of traceRouteCtlInitialTtl for traceRouteProbeHistoryHopIndex.
- The fifth variable, traceRouteProbeHistoryProbeIndex, is the probe for the current hop. It ranges from 1 to traceRouteCtlProbesPerHop.

While a test is running, as soon as a probe result is determined, the next probe is sent. A maximum of traceRouteCtlTimeOut seconds elapses before a probe is marked with status requestTimedOut and the next probe is sent. There is never more than one outstanding probe per traceroute test. Any probe result coming back after a probe times out is ignored.

Each probe can:

- Result in a response from a host acknowledging the probe
- Time out with no response from a host acknowledging the probe
- Fail to be sent

Each probe status is recorded in traceRouteProbeHistoryTable with traceRouteProbeHistoryStatus set accordingly.

Probes that result in a response from a host record the following data:

- traceRouteProbeHistoryResponse—Round-trip time (RTT)
- traceRouteProbeHistoryHAddrType—The type of HAddr (next argument)
- traceRouteProbeHistoryHAddr—The address of the hop

All probes, regardless of whether a response for the probe is received, have the following recorded:

- traceRouteProbeHistoryStatus—What happened and why

- traceRouteProbeHistoryLastRC—Return code (RC) value of the ICMP packet
- traceRouteProbeHistoryTime—Timestamp when the probe result was determined

When a probe cannot be sent, traceRouteProbeHistoryResponse is set to 0. When a probe times out, traceRouteProbeHistoryResponse 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.

traceRouteHopsTable

Entries in traceRouteHopsTable are indexed by three variables:

- The first two, traceRouteCtlOwnerIndex and traceRouteCtlTestName, are the same ones used for traceRouteCtlTable and identify the test.
- The third variable, traceRouteHopsHopIndex, indicates the current hop, which starts at 1 (not traceRouteCtlInitialTtl).

When a test starts, all entries in traceRouteHopsTable with the given traceRouteCtlOwnerIndex and traceRouteCtlTestName are deleted. Entries in this table are only created if traceRouteCtlCreateHopsEntries is set to true.

A new traceRouteHopsEntry is created each time the first probe result for a given TTL is determined. The new entry is created whether or not the first probe reaches a host. The value of traceRouteHopsHopIndex is increased by 1 for this new entry.

NOTE: Any traceRouteHopsEntry can lack a value for traceRouteHopsIpTgtAddress if there are no responses to the probes with the given TTL.

Each time a probe reaches a host, the IP address of that host is available in the probe result. If the value of traceRouteHopsIpTgtAddress of the current traceRouteHopsEntry is not set, then the value of traceRouteHopsIpTgtAddress is set to this IP address. If the value of traceRouteHopsIpTgtAddress of the current traceRouteHopsEntry is the same as the IP address, then the value does not change. If the value of traceRouteHopsIpTgtAddress of the current traceRouteHopsEntry is different from this IP address, indicating a path change, a new traceRouteHopsEntry is created with:

- traceRouteHopsHopIndex variable increased by 1
- traceRouteHopsIpTgtAddress set to the IP address

NOTE: A new entry for a test is added to traceRouteHopsTable each time a new TTL value is used or the path changes. Thus, the number of entries for a test may exceed the number of different TTL values used.

When a probe result is determined, the value traceRouteHopsSentProbes of the current traceRouteHopsEntry increases by 1. When a probe result is determined, and the probe reaches a host:

- The value traceRouteHopsProbeResponses of the current traceRouteHopsEntry is increased by 1.
- The following variables are updated:
  - traceRouteResultsMinRtt—Minimum round-trip time
  - traceRouteResultsMaxRtt—Maximum round-trip time
  - traceRouteResultsAverageRtt—Average round-trip time
  - traceRouteResultsRttSumOfSquares—Sum of squares of round-trip times
  - traceRouteResultsLastGoodProbe—Timestamp of the last response

> **NOTE**: Only probes that reach a host affect the round-trip time values.

Generate Traps

To generate any trap, an appropriate bit of traceRouteCtlTrapGeneration must be set. You must also configure a trap group to receive remote operations. Traps are generated under the following conditions:

- traceRouteHopsIpTgtAddress of the current probe is different from the last probe with the same TTL value (traceRoutePathChange).
- A path to the target could not be determined (traceRouteTestFailed).

A path to the target was determined (traceRouteTestCompleted).

For information about how to configure a trap group to receive remote operations, see *Configuring SNMP Trap Groups* and "SNMP Remote Operations Overview" on page 432.

Monitor Traceroute Test Completion

When a test is complete, `traceRouteResultsOperStatus` transitions from `enabled` to `disabled`. This transition occurs in the following situations:

- The test ends successfully. A probe result indicates that the destination has been reached. In this case, the current hop is the last hop. The rest of the probes for this hop are sent. When the last probe result for the current hop is determined, the test ends.

- traceRouteCtlMaxTtl threshold is exceeded. The destination is never reached. The test ends after the number of probes with TTL value equal to traceRouteCtlMaxttl have been sent.
- traceRouteCtlMaxFailures threshold is exceeded. The number of consecutive probes that end with status requestTimedOut exceeds traceRouteCtlMaxFailures.
- You end the test. You set traceRouteCtlAdminStatus to disabled or delete the row by setting traceRouteCtlRowStatus to destroy.
- You misconfigured the traceroute test. A value or variable you specified in traceRouteCtlTable is incorrect and will not allow a single probe to be sent. Because of the nature of the data, this error could not be determined until the test was started; that is, until after traceRouteResultsOperStatus transitioned to enabled. When this occurs, one entry is added to traceRouteProbeHistoryTable with traceRouteProbeHistoryStatus set to the appropriate error code.

If traceRouteCtlTrapGeneration is set properly, either the traceRouteTestFailed or traceRouteTestCompleted trap is generated.

Gather Traceroute Test Results

You can either poll traceRouteResultsOperStatus to find out when the test is complete or request that a trap be sent when the test is complete. For more information about traceResultsOperStatus, see "traceRouteResultsTable" on page 446. For more information about Traceroute MIB traps, see the Generating Traps section in "Monitoring a Running Traceroute Test" on page 446.

Statistics are calculated on a per-hop basis and then stored in traceRouteHopsTable. They include the following for each hop:

- traceRouteHopsIpTgtAddressType—Address type of host at this hop
- traceRouteHopsIpTgtAddress—Address of host at this hop
- traceRouteHopsMinRtt—Minimum round-trip time
- traceRouteHopsMaxRtt—Maximum round-trip time
- traceRouteHopsAverageRtt—Average round-trip time
- traceRouteHopsRttSumOfSquares—Sum of squares of round-trip times
- traceRouteHopsSentProbes—Number of attempts to send probes
- traceRouteHopsProbeResponses—Number of responses received
- traceRouteHopsLastGoodProbe—Timestamp of last response

You can also consult traceRouteProbeHistoryTable for more detailed information about each probe. The index used for traceRouteProbeHistoryTable starts at 1, goes to 0xFFFFFFFF, and wraps to 1 again.

For example, assume the following:

- traceRouteCtlMaxRows is 10.
- traceRouteCtlProbesPerHop is 5.
- There are eight hops to the target (the target being number eight).
- Each probe sent results in a response from a host (the number of probes sent is not limited by traceRouteCtlMaxFailures).

In this test, 40 probes are sent. At the end of the test, traceRouteProbeHistoryTable would have a history of probes like those in Table 36 on page 452.

Table 36: traceRouteProbeHistoryTable

HistoryIndex	HistoryHopIndex	HistoryProbeIndex
31	7	1
32	7	2
33	7	3
34	7	4
35	7	5
36	8	1
37	8	2
38	8	3
39	8	4

Table 36: traceRouteProbeHistoryTable *(Continued)*

HistoryIndex	HistoryHopIndex	HistoryProbeIndex
40	8	5

Stop a Traceroute Test

To stop an active test, set `traceRouteCtlAdminStatus` to `disabled`. To stop a test and remove its `traceRouteCtlEntry`, `traceRouteResultsEntry`, `traceRouteProbeHistoryEntry`, and `traceRouteProbeHistoryEntry` objects from the MIB, set `traceRouteCtlRowStatus` to `destroy`.

Interpret Traceroute Variables

This topic contains information about the ranges for the following variables that are not explicitly specified in the Traceroute MIB:

- `traceRouteCtlMaxRows`—The maximum value for `traceRouteCtlMaxRows` is 2550. This represents the maximum TTL (255) multiplied by the maximum for `traceRouteCtlProbesPerHop` (10). Therefore, the `traceRouteProbeHistoryTable` accommodates one complete test at the maximum values for one `traceRouteCtlEntry`. Usually, the maximum values are not used and the `traceRouteProbeHistoryTable` is able to accommodate the complete history for many tests for the same `traceRouteCtlEntry`.
- `traceRouteMaxConcurrentRequests`—The maximum value is 50. If a test is running, it has one outstanding probe. `traceRouteMaxConcurrentRequests` represents the maximum number of traceroute tests that have `traceRouteResultsOperStatus` with a value of `enabled`. Any attempt to start a test with `traceRouteMaxConcurrentRequests` tests running will result in the creation of one probe with `traceRouteProbeHistoryStatus` set to `maxConcurrentLimitReached` and that test will end immediately.
- `traceRouteCtlTable`—The maximum number of entries allowed in this table is 100. Any attempt to create a 101st entry will result in a `BAD_VALUE` message for SNMPv1 and a `RESOURCE_UNAVAILABLE` message for SNMPv2.

Release History Table

Release	Description
17.2X75-D100	Starting in Junos OS Release 17.2X75-D100, you must configure RPM before starting an ICMP ping.

SNMP Traps

IN THIS SECTION

- Configure SNMP Traps | **454**
- Configure SNMP Trap Options | **456**
- Configure SNMP Trap Groups | **461**
- Configure SNMP Trap Options and Groups on a Device Running Junos OS | **463**
- Example: Configure SNMP Trap Groups | **464**
- Manage Traps | **465**

Configure SNMP Traps

Traps are unsolicited messages sent from an SNMP agent to remote network management systems, or trap receivers. Enterprises use SNMP traps as part of a fault-monitoring solution in addition to system logging. In Junos OS, you must configure a trap-group if you wish to use SNMP traps.

You can create and name a group of one or more types of SNMP traps and define which systems receive the group of SNMP traps. The name of the trap group is embedded in SNMP trap notification packets as one variable binding (varbind) known as the community name.

To configure an SNMP trap:

1. Create a single, consistent source address that Junos OS applies to all outgoing traps in your device.

A source address is useful, because although most Junos OS devices have several outbound interfaces, using one source address helps a remote NMS to associate the source of the traps with an individual device

```
[edit groups global snmp]
user@host# set trap-options source-address address
```

This example uses the IP address of the loopback interface (lo0) as the source address for all the SNMP traps that originate from the device.

```
[edit groups global snmp]
user@host# set trap-options source-address lo0
```

2. Create a trap group in which you can list the types of traps to be forwarded and the targets (addresses) of the receiving remote management systems.

```
[edit groups global snmp trap-group group-name]
user@host# set version (all | v1 | v2) targets address
```

This example creates a trap group called `managers`, allows SNMP version 2-formatted notifications (traps) to be sent to the host at address 192.168.1.15. This statement forwards all categories of traps.

```
[edit groups global snmp trap-group managers]
user@host# set version v2 targets 192.168.1.15
```

3. Define the specific subset of trap categories to be forwarded.

   For a list of categories, see Configure SNMP Trap Groups.

```
[edit groups global snmp trap-group group-name]
user@host# set categories category
```

The following statement configures the standard MIB-II authentication failures on the agent (the device).

```
[edit groups global snmp trap-group managers]
user@host# set categories authentication
```

4. At the top level of the configuration, apply the configuration group.

If you use a configuration group, you must apply it for it to take effect.

```
[edit]
user@host# set apply-groups global
```

5. Commit the configuration.

```
user@host# commit
```

6. To verify the configuration, generate an authentication failure trap.

   This means that the SNMP agent received a request with an unknown community. Other traps types can also be spoofed as well.

   This feature enables you to trigger SNMP traps from routers and ensure that they are processed correctly within your existing network management infrastructure. This is also useful for testing and debugging SNMP behavior on the switch or NMS.

   Using the `monitor traffic` command, you can verify that the trap is sent to the network management system.

```
user@host> request snmp spoof-trap authenticationFailure
Spoof-trap request result: trap sent successfully
```

Configure SNMP Trap Options

IN THIS SECTION

- Configure the Source Address for SNMP Traps | **457**
- Configure the Agent Address for SNMP Traps | **460**
- Add snmpTrapEnterprise Object Identifier to Standard SNMP Traps | **460**

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: You can associate SNMP with only master routing instance.

To configure SNMP trap options, see *trap-options*.

You must also configure a trap group for the trap options to take effect. For information about trap groups, see Configure SNMP Trap Groups.

This topic contains the following sections:

Configure the Source Address for SNMP Traps

You can configure the source address of trap packets in many ways: lo0, a valid IPv4 address or IPv6 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.

NOTE: You can generate SNMP Traps only if the source address is a valid IPv4 or IPv6 address or is configured.

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
- A valid IPv6 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 IPv4 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.

A Valid IPv6 Address As the Source Address

To specify a valid IPv6 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 IPv6 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;
}
```

Configure 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 not specified 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.

Add 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;
}
```

Configure SNMP Trap Groups

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. You must configure the trap group for sending the SNMP traps. To create an SNMP trap group, see *trap-group*.

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 by Junos OS and Enterprise-Specific SNMP Traps Supported by Junos OS 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
- `chassis-cluster`—Clustering 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.

- otn-alarms—OTN alarm trap subcategories
- remote-operations—Remote operation notifications
- rmon-alarm—Alarm for RMON events
- routing—Routing protocol notifications
- services—Services notifications such as circuit down or up, connection down or up, CPU exceeded, alarms, and status changes.
- 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-primary 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 (SNMP)*.

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

Some carriers have more than one trap receiver that forwards traps to a central NMS. This allows more than one path for SNMP traps from a router to the central NMS through different trap receivers. You can configure a device running Junos OS 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 recognise the duplicate traps and 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);
}
```

Example: Configure SNMP Trap Groups

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;
        }
    }
}
```

Manage Traps

The following provide details on managing SNMP notifications:

- **Generate 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:

  The following example shows the sample configuration for raising a trap for the event `ui_mgd_terminate`:

```
[edit event-options policy p1]
events ui_mgd_terminate;
then {
    raise-trap;
}
```

- **Filter 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;
    }
}
```

- **Filter 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;
    }
}
```

SNMP Traps Supported by Junos OS

IN THIS SECTION

- SNMP Traps Support | **468**
- Standard SNMP Traps Supported by Junos OS | **487**
- Customized SNMP MIBs for Syslog Traps | **498**

The QFX Series standalone switches, QFX Series Virtual Chassis, and QFabric systems support standard SNMP traps and Juniper Networks enterprise-specific traps.

SNMP Traps Support

IN THIS SECTION

- SNMP Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis | 468
- SNMP Traps Supported on QFabric Systems | 482

SNMP Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

QFX Series standalone switches and QFX Series Virtual Chassis support SNMPv1 and v2 traps. For more information, see:

SNMPv1 Traps

QFX Series standalone switches and QFX Series Virtual Chassis support both standard SNMPv1 traps and Juniper Networks enterprise-specific SNMPv1 traps. See:

- Table 37 on page 469 for standard SNMPv1 traps.
- Table 38 on page 472 for enterprise-specific SNMPv1 traps.

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 (–).

Table 37: Standard SNMP Version 1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag
Link Notifications						
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
	linkUp	1.3.6.1.4.1.2636	3	0	Info	SNMP_TRAP_ LINK_UP
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
	pingTestFailed	1.3.6.1.2.1.80.0	6	2	Info	SNMP_TRAP_ PING_TEST _FAILED
	pingTestCompleted	1.3.6.1.2.1.80.0	6	3	Info	SNMP_TRAP_ PING_TEST_ COMPLETED
	traceRoutePathChange	1.3.6.1.2.1.81.0	6	1	Info	SNMP_TRAP_ TRACE_ROUTE_ PATH_CHANGE
	traceRouteTestFailed	1.3.6.1.2.1.81.0	6	2	Info	SNMP_TRAP_ TRACE_ROUTE_ TEST_FAILED

Table 37: Standard SNMP Version 1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag
	traceRouteTestCompleted	1.3.6.1.2.1.81.0	6	3	Info	SNMP_TRAP_TRACE_ROUTE_TEST_COMPLETED
RMON Alarms						
RFC 2819a, *RMON MIB*	fallingAlarm	1.3.6.1.2.1.16	6	2	–	–
	risingAlarm	1.3.6.1.2.1.16	6	1	–	–
Routing Notifications						
BGP 4 MIB	bgpEstablished	1.3.6.1.2.1.15.7	6	1	–	–
	bgpBackwardTransition	1.3.6.1.2.1.15.7	6	2	–	–
OSPF TRAP MIB	ospfVirtIfStateChange	1.3.6.1.2.1.14.16.2	6	1	–	–
	ospfNbrStateChange	1.3.6.1.2.1.14.16.2	6	2	–	–
	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2	6	3	–	–
	ospfIfConfigError	1.3.6.1.2.1.14.16.2	6	4	–	–

Table 37: Standard SNMP Version 1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag
	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2	6	5	–	–
	ospfIfAuthFailure	1.3.6.1.2.1.14.16.2	6	6	–	–
	ospfVirtIfAuthFailure	1.3.6.1.2.1.14.16.2	6	7	–	–
	ospfIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	8	–	–
	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	9	–	–
	ospfTxRetransmit	1.3.6.1.2.1.14.16.2	6	10	–	–
	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2	6	11	–	–
	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2	6	13	–	–
	ospfIfStateChange	1.3.6.1.2.1.14.16.2	6	16	–	–
Startup Notifications						

Table 37: Standard SNMP Version 1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	Syslog Tag
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
	coldStart	1.3.6.1.4.1.2636	0	0	Critical	SNMPD_TRAP_ COLD_START
	warmStart	1.3.6.1.4.1.2636	1	0	Error	SNMPD_TRAP_ WARM_START
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
	vrrpTrapAuthFailure	1.3.6.1.2.1.68	6	2	Warning	VRRPD_AUTH_ FAILURE_TRAP

Table 38: Enterprise-Specific SNMPv1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag
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

Table 38: Enterprise-Specific SNMPv1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag
	jnxFanFailure	1.3.6.1.4.1.2636 1	6	2	Critical	CHASSISD_ SNMP_ TRAP
	jnxOverTemperature	11.4.1.2636.4.1	6	3	Alert	CHASSISD_ SNMP_ TRAP
	jnxFruRemoval	1.3.6.1.4.1.2636 .4.1	6	5	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruInsertion	1.3.6.1.4.1.2636 .4.1	6	6	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruPowerOff	1.3.6.1.4.1.2636 .4.1	6	7	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruPowerOn	1.3.6.1.4.1.2636 .4.1	6	8	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruFailed	1.3.6.1.4.1.2636 .4.1	6	9	Warning	CHASSISD_ SNMP_ TRAP
	jnxFruOffline	1.3.6.1.4.1.2636 .4.1	6	10	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruOnline	1.3.6.1.4.1.2636 .4.1	6	11	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruCheck	1.3.6.1.4.1.2636 .4.1	6	12	Warning	CHASSISD_ SNMP_ TRAP

Table 38: Enterprise-Specific SNMPv1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag
	jnxPowerSupplyOk	1.3.6.1.4.1.2636.4.2	6	1	Critical	CHASSISD_SNMP_TRAP
	jnxFanOK	1.3.6.1.4.1.2636.4.2	6	2	Critical	CHASSISD_SNMP_TRAP
	jnxTemperatureOK	1.3.6.1.4.1.2636.4.2	6	3	Alert	CHASSISD_SNMP_TRAP
Configuration Notifications						
Configuration Management MIB (jnx-configmgmt. mib)	jnxCmCfgChange	1.3.6.1.4.1.2636.4.5	6	1	–	–
	jnxCmRescueChange	1.3.6.1.4.1.2636.4.5	6	2	–	–
Remote Operations						
Ping MIB (jnx-ping.mib)	jnxPingRttThresholdExceeded	1.3.6.1.4.1.2636.4.9	6	1	–	–
	jnxPingRttStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	2	–	–
	jnxPingRttJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	3	–	–
	jnxPingEgressThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	4	–	–

Table 38: Enterprise-Specific SNMPv1 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	Enterprise ID	Generic Trap Number	Specific Trap Number	System Logging Severity Level	System Log Tag
	jnxPingEgressStdDevThresholdExceeded	1.3.6.1.4.1.2636.4.9	6	5	–	–
	jnxPingEgressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	6	–	–
	jnxPingIngressThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	7	–	–
	jnxPingIngressStddevThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	8	–	–
	jnxPingIngressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9	6	9	–	–
RMON Alarms						
RMON MIB (jnx-rmon. mib)	jnxRmonAlarmGetFailure	1.3.6.1.4.1.2636.4.3	6	1	–	–
	jnxRmonGetOk	1.3.6.1.4.1.2636.4.3	6	2	–	–

SNMPv2 Traps

- Table 39 on page 476 lists the standard SNMP traps
- Table 40 on page 479 lists the Juniper Networks enterprise-specific traps

Table 39: Standard SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

Defined in	Trap Name	SNMP Trap OID	System Logging Severity Level	Syslog Tag
Link Notifications				
RFC 2863, *The Interfaces Group MIB*	linkDown	1.3.6.1.6.3.1.1.5.3	Warning	SNMP_TRAP_LINK_DOWN
	linkUp	1.3.6.1.6.3.1.1.5.4	Info	SNMP_TRAP_LINK_UP
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
	pingTestFailed	1.3.6.1.2.1.80.0.2	Info	SNMP_TRAP_PING_TEST_FAILED
	pingTestCompleted	1.3.6.1.2.1.80.0.3	Info	SNMP_TRAP_PING_TEST_COMPLETED
	traceRoutePathChange	1.3.6.1.2.1.81.0.1	Info	SNMP_TRAP_TRACE_ROUTE_PATH_CHANGE
	traceRouteTestFailed	1.3.6.1.2.1.81.0.2	Info	SNMP_TRAP_TRACE_ROUTE_TEST_FAILED
	traceRouteTestCompleted	1.3.6.1.2.1.81.0.3	Info	SNMP_TRAP_TRACE_ROUTE_TEST_COMPLETED
RMON Alarms				

Table 39: Standard SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	SNMP Trap OID	System Logging Severity Level	Syslog Tag
RFC 2819a, *RMON MIB*	fallingAlarm	1.3.6.1.2.1.16.0.1	–	–
	risingAlarm	1.3.6.1.2.1.16.0.2	–	–
Routing Notifications				
BGP 4 MIB	bgpEstablished	1.3.6.1.2.1.15.7.1	–	–
	bgpBackwardTransition	1.3.6.1.2.1.15.7.2	–	–
OSPF Trap MIB	ospfVirtIfStateChange	1.3.6.1.2.1.14.16.2.1	–	–
	ospfNbrStateChange	1.3.6.1.2.1.14.16.2.2	–	–
	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2.3	–	–
	ospfIfConfigError	1.3.6.1.2.1.14.16.2.4	–	–
	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2.5	–	–
	ospfIfAuthFailure	1.3.6.1.2.1.14.16.2.6	–	–

Table 39: Standard SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	SNMP Trap OID	System Logging Severity Level	Syslog Tag
	ospfVirtIfAuthFailure	1.3.6.1.2.1.14.16.2.7	–	–
	ospfIfRxBadPacket	1.3.6.1.2.1.14.16.2.8	–	–
	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2.9	–	–
	ospfTxRetransmit	1.3.6.1.2.1.14.16.2.10	–	–
	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2.11	–	–
	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2.13	–	–
	ospfIfStateChange	1.3.6.1.2.1.14.16.2.16	–	–
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
	warmStart	1.3.6.1.6.3.1.1.5.2	Error	SNMPD_TRAP_WARM_START
	authenticationFailure	1.3.6.1.6.3.1.1.5.5	Notice	SNMPD_TRAP_GEN_FAILURE

Table 39: Standard SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Defined in	Trap Name	SNMP Trap OID	System Logging Severity Level	Syslog Tag
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
	vrrpTrapAuthFailure	1.3.6.1.2.1.68.0.2	Warning	VRRPD_AUTH_ FAILURE_ TRAP

Table 40: Enterprise-Specific SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
Chassis (Alarm Conditions) Notifications				
Chassis MIB (mib-jnx-chassis)	jnxPowerSupplyFailure	1.3.6.1.4.1.2636.4.1.1	Alert	CHASSISD_ SNMP_ TRAP
	jnxFanFailure	1.3.6.1.4.1.2636.4.1.2	Critical	CHASSISD_ SNMP_ TRAP
	jnxOverTemperature	1.3.6.1.4.1.2636.4.1.3	Critical	CHASSISD_ SNMP_ TRAP
	jnxFruRemoval	1.3.6.1.4.1.2636.4.1.5	Notice	CHASSISD_ SNMP_ TRAP

Table 40: Enterprise-Specific SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
	jnxFruInsertion	1.3.6.1.4.1.2636.4.1.6	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruPowerOff	1.3.6.1.4.1.2636.4.1.7	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruPowerOn	1.3.6.1.4.1.2636.4.1.8	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruFailed	1.3.6.1.4.1.2636.4.1.9	Warning	CHASSISD_ SNMP_ TRAP
	jnxFruOffline	1.3.6.1.4.1.2636.4.1.10	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruOnline	1.3.6.1.4.1.2636.4.1.11	Notice	CHASSISD_ SNMP_ TRAP
	jnxFruCheck	1.3.6.1.4.1.2636.4.1.12	Notice	CHASSISD_ SNMP_ TRAP
	jnxPowerSupplyOK	1.3.6.1.4.1.2636.4.2.1	Critical	CHASSISD_ SNMP_ TRAP
	jnxFanOK	1.3.6.1.4.1.2636.4.2.2	Critical	CHASSISD_ SNMP_ TRAP
	jnxTemperatureOK	1.3.6.1.4.1.2636.4.2.3	Alert	CHASSISD_ SNMP_ TRAP

Table 40: Enterprise-Specific SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
Configuration Notifications				
Configuration Management MIB (mib-jnx-cfgmgmt)	jnxCmCfgChange	1.3.6.1.4.1.2636.4.5.0.1	–	–
	jnxCmRescueChange	1.3.6.1.4.1.2636.4.5.0.2	–	–
Remote Operations Notifications				
Ping MIB (mib-jnx-ping)	jnxPingRttThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.1	–	–
	jnxPingRttStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.2	–	–
	jnxPingRttJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.3	–	–
	jnxPingEgressThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.4	–	–
	jnxPingEgressStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.5	–	–
	jnxPingEgressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.6	–	–
	jnxPingIngressThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.7	–	–

Table 40: Enterprise-Specific SNMPv2 Traps Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
	jnxPingIngressStddevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.8	–	–
	jnxPingIngressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.9	–	–
RMON Alarms				
RMON MIB (mib-jnx-rmon)	jnxRmonAlarmGetFailure	1.3.6.1.4.1.2636.4. 3.0.1	–	–
	jnxRmonGetOk	1.3.6.1.4.1.2636.4. 3.0.2	–	–

SNMP Traps Supported on QFabric Systems

QFabric systems support standard SNMPv2 traps and Juniper Networks enterprise-specific SNMPv2 traps.

NOTE: QFabric systems do not support SNMPv1 traps.

For more information, see:

- Table 41 on page 483 for standard SNMPv2 traps
- Table 42 on page 483 for Juniper Networks enterprise-specific SNMPv2 traps

Table 41: Standard SNMPv2 Traps Supported on QFabric Systems

Defined in	Trap Name	SNMP Trap OID	System Logging Severity Level	Syslog Tag
Link Notifications				
RFC 2863, *The Interfaces Group MIB*	linkDown	1.3.6.1.6.3.1.1.5.3	Warning	SNMP_TRAP_LINK_DOWN
	linkUp	1.3.6.1.6.3.1.1.5.4	Info	SNMP_TRAP_LINK_UP
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
	warmStart	1.3.6.1.6.3.1.1.5.2	Error	SNMPD_TRAP_WARM_START
	authenticationFailure	1.3.6.1.6.3.1.1.5.5	Notice	SNMPD_TRAP_GEN_FAILURE

Table 42: Enterprise-Specific SNMPv2 Traps Supported on QFabric Systems

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
Fabric Chassis MIB (mib-jnx-fabric- chassis)	**Fabric Chassis (Alarm Conditions) Notifications**			
	jnxFabricPowerSupplyFailure	1.3.6.1.4.1.2636.4.19.1	Warning	–
	jnxFabricFanFailure	1.3.6.1.4.1.2636.4.19.2	Critical	–

Table 42: Enterprise-Specific SNMPv2 Traps Supported on QFabric Systems *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
	jnxFabricOverTemperature	1.3.6.1.4.1.2636.4.19.3	Alert	–
	jnxFabricRedundancySwitchover	1.3.6.1.4.1.2636.4.19.4	Notice	–
	jnxFabricFruRemoval	1.3.6.1.4.1.2636.4.19.5	Notice	–
	jnxFabricFruInsertion	1.3.6.1.4.1.2636.4.19.6	Notice	–
	jnxFabricFruPowerOff	1.3.6.1.4.1.2636.4.19.7	Notice	–
	jnxFabricFruPowerOn	1.3.6.1.4.1.2636.4.19.8	Notice	–
	jnxFabricFruFailed	1.3.6.1.4.1.2636.4.19.9	Warning	–
	jnxFabricFruOffline	1.3.6.1.4.1.2636.4.19.10	Notice	–
	jnxFabricFruOnline	1.3.6.1.4.1.2636.4.19.11	Notice	–
	jnxFabricFruCheck	1.3.6.1.4.1.2636.4.19.12	Warning	–
	jnxFabricFEBSwitchover	1.3.6.1.4.1.2636.4.19.13	Warning	–
	jnxFabricHardDiskFailed	1.3.6.1.4.1.2636.4.19.14	Warning	–
	jnxFabricHardDiskMissing	1.3.6.1.4.1.2636.4.19.15	Warning	–
	jnxFabricBootFromBackup	1.3.6.1.4.1.2636.4.19.16	Warning	–

Table 42: Enterprise-Specific SNMPv2 Traps Supported on QFabric Systems *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
	Fabric Chassis (Alarm Cleared Conditions) Notifications			
	jnxFabricPowerSupplyOK	1.3.6.1.4.1.2636.4.20.1	Critical	–
	jnxFabricFanOK	1.3.6.1.4.1.2636.4.20.2	Critical	–
	jnxFabricTemperatureOK	1.3.6.1.4.1.2636.4.20.3	Alert	–
	jnxFabricFruOK	1.3.6.1.4.1.2636.4.20.4	–	–
QFabric MIB (mib-jnx-qf-smi)	**QFabric MIB Notifications**			
	jnxQFabricDownloadIssued	1.3.6.1.4.1.2636.3.42.1.0.1	–	–
	jnxQFabricDownloadFailed	1.3.6.1.4.1.2636.3.42.1.0.2	–	–
	jnxQFabricDownloadSucceeded	1.3.6.1.4.1.2636.3.42.1.0.3	–	–
	jnxQFabricUpgradeIssued	1.3.6.1.4.1.2636.3.42.1.0.4	–	–
	jnxQFabricUpgradeFailed	1.3.6.1.4.1.2636.3.42.1.0.5	–	–
	jnxQFabricUpgradeSucceeded	1.3.6.1.4.1.2636.3.42.1.0.6	–	–
Configuration Notifications				

Table 42: Enterprise-Specific SNMPv2 Traps Supported on QFabric Systems *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
Configuration Management MIB (mib-jnx-cfgmgmt)	jnxCmCfgChange	1.3.6.1.4.1.2636.4.5.0.1	–	–
	jnxCmRescueChange	1.3.6.1.4.1.2636.4.5.0.2	–	–
Remote Operations Notifications				
Ping MIB (mib-jnx-ping)	jnxPingRttThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.1	–	–
	jnxPingRttStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.2	–	–
	jnxPingRttJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.3	–	–
	jnxPingEgressThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.4	–	–
	jnxPingEgressStdDevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.5	–	–
	jnxPingEgressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.6	–	–
	jnxPingIngressThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.7	–	–
	jnxPingIngressStddevThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.8	–	–

Table 42: Enterprise-Specific SNMPv2 Traps Supported on QFabric Systems *(Continued)*

Source MIB	Trap Name	SNMP Trap OID	System Logging Severity Level	System Log Tag
	jnxPingIngressJitterThreshold Exceeded	1.3.6.1.4.1.2636.4.9.0.9	–	–

SEE ALSO

SNMP MIB Explorer

No Link Title

Understanding the Implementation of SNMP on the QFabric System

SNMP MIBs Support

Standard SNMP Traps Supported by Junos OS

IN THIS SECTION

- Standard SNMP Version 1 Traps | **488**
- Standard SNMP Version 2 Traps | **492**

This topic provides the list of standard SNMPv1 and SNMPv2 traps supported by devices running Junos OS. For more information about traps see SNMP MIB Explorer.

Starting in Junos OS Release 20.1, after graceful routing engine switchover (GRES), the new primary Routing Engine sends a single warmStart notification. The primary Routing Engine sends a `coldStart` notification when the device comes up. The primary Routing Engine also sends `warmStart` notifications for subsequent restarts of the SNMP daemon. After GRES, the new primary Routing Engine sends a single `warmStart` notification and the backup Routing Engine does not send any notification.

Standard SNMP Version 1 Traps

Table 43 on page 488 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 System Log Explorer.

Table 43: 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							
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							

Table 43: 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 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.
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 Series Firewalls.
	bgpBackwardTransition	1.3.6.1.2.1.15.7	6	2	–	–	M, T, MX, J, EX, and SRX Series Firewalls.

Table 43: 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 Series Firewalls.
	ospfNbrStateChange	1.3.6.1.2.1.14.16.2	6	2	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfVirtNbrStateChange	1.3.6.1.2.1.14.16.2	6	3	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfIfConfigError	1.3.6.1.2.1.14.16.2	6	4	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfVirtIfConfigError	1.3.6.1.2.1.14.16.2	6	5	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfIfAuthFailure	1.3.6.1.2.1.14.16.2	6	6	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfVirtIfAuthFailure	1.3.6.1.2.1.14.16.2	6	7	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	8	–	–	M, T, MX, J, EX, and SRX Series Firewalls.

Table 43: 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
	ospfVirtIfRxBadPacket	1.3.6.1.2.1.14.16.2	6	9	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfTxRetransmit	1.3.6.1.2.1.14.16.2	6	10	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfVirtIfTxRetransmit	1.3.6.1.2.1.14.16.2	6	11	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfMaxAgeLsa	1.3.6.1.2.1.14.16.2	6	13	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
	ospfIfStateChange	1.3.6.1.2.1.14.16.2	6	16	–	–	M, T, MX, J, EX, and SRX Series Firewalls.
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.

Table 43: 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 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

Standard SNMP Version 2 Traps

Table 44 on page 492 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.

Table 44: Standard Supported SNMP Version 2 Traps

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

Table 44: Standard Supported SNMP Version 2 Traps *(Continued)*

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

Table 44: Standard Supported SNMP Version 2 Traps *(Continued)*

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

Table 44: Standard Supported SNMP Version 2 Traps *(Continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
	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					
RFC 3812, *Multiprotocol Label Switching (MPLS) Traffic*	mplsTunnelUp				

Table 44: Standard Supported SNMP Version 2 Traps *(Continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
Engineering (TE) Management Information Base	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				
	mplsL3VpnVrfNumVrfRouteMaxThreshExceeded				

Table 44: Standard Supported SNMP Version 2 Traps *(Continued)*

Defined in	Trap Name	snmpTrapOID	System Logging Severity Level	Syslog Tag	Supported On
	`mplsL3VpnNum VrfRouteMax ThreshCleared`				
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_MA STER	M and MX
	`vrrpv3ProtoError`	1.3.6.1.2.1.207. 0.2	Warning	VRRPD_V3_PROT O_ERROR	M and MX

SEE ALSO

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

No Link Title

Customized SNMP MIBs for Syslog Traps

IN THIS SECTION

- Overview of Custom SNMP MIBs | 498
- Define a Custom MIB for a Syslog Trap | 500
- Limitations of Using Custom SNMP Traps | 507
- Example Custom Syslog Trap | 507

SNMP syslog traps are alert messages sent from a remote SNMP-enabled device to a central collector notifying you of a component failure or when critical resources are out of configurable limits. This information is captured in a Management Information Base (MIB). The Juniper Networks enterprise-specific System Log MIB enables notification of an SNMP trap-based application when an important system log message occurs. The MIB is defined to map the syslog entry to the generic jnxSyslogTrap OID.

The jnxSyslogTrap OID is a trap based on the logs generated in the syslog. The Event process (eventd) monitors syslog and, based on the event policy `raise-trap` configuration statement for syslog events, sends all syslog events into one generic syslog-defined trap MIB, which is jnxSyslogTrap.

Using one generic MIB OID is inconvenient for customers who want to process syslog trap OID values to discover specific events because it is impossible to distinguish alarms having the same OID. But as of Junos OS Release 18.3R1, you can map a custom OID to a particular log and load it on the device dynamically.

The benefit of this feature is that because there is a way to assign specific OIDs to different types of syslog events, you can now effectively monitor for each different type of syslog event.

Overview of Custom SNMP MIBs

IN THIS SECTION

- Write the MIB File | 499
- Convert to a YANG File | 499
- CLI Commands to Use for Managing YANG Files | 500

To create a custom SNMP MIB for a syslog trap, you must complete the following tasks:

- Write the custom MIB.
- Convert the MIB file to YANG format and copy the YANG file to the device.
- Load the YANG file onto the device.

The following sections overview these steps.

Write the MIB File

Before you can map a particular log with a custom OID, you must write a custom MIB. To avoid collisions, you must define your MIB objects and traps only under the reserved roots shown in Table 9.

Table 45: MIB Roots for Custom MIB Modules

Root	Description	OID
`.iso.org.dod.internet.private.enterprises.juniperMIB.jnxMibs.jnxCustomMibRoot`	Custom MIB module	.1.3.6.1.4.1.2636.3.86
`.iso.org.dod.internet.private.enterprises.juniperMIB.jnxTraps.jnxCustomSyslogNotifications`	Custom trap notification	.1.3.6.1.4.1.2636.4.30

Convert to a YANG File

Before loading your MIB definition onto the device, you must convert the MIB file to YANG format. The recommended way to covert the MIB file to YANG is to use the smidump v0.5.0 tool. The smidump tool is an open source application which can be installed on your laptop (see https://www.ibr.cs.tu-bs.de/projects/libsmi/smidump.html).

Once the file is in YANG format, you must copy it to the device. Then, using a CLI command, you load the into the SNMP process (snmpd). A corresponding JSON file is then generated, which snmpd parses and from it builds the database of the OID hierarchy. If some unknown tag is found, snmpd returns the appropriate error message.

CLI Commands to Use for Managing YANG Files

To load the YANG module into snmpd, use the `snmp` option with the `request system yang add` command:

```
user@host> request system yang add snmp module yang-filename package package-name
```

The *yang-filename* includes the absolute path.

NOTE: In order to run the `request system yang add` command, you must have super-user access.

There are two other commands for managing YANG files on devices: `show system yang package` and `request system yang delete`.

SEE ALSO

show system yang package

request system yang delete

request system yang add

Define a Custom MIB for a Syslog Trap

In this procedure, we use the following example files:

- MIB file to convert
- output

NOTE: Although YANG can be written manually by referring to the example YANG provided in this documentation, we recommend you convert the MIB to YANG format using the smidump tool v0.5.0.

To define a custom MIB for a syslog trap:

1. Load your MIB onto the network management system (NMS) and check if there are any errors.
2. Invoke the smidump tool using the following command, where ***dependency-mib***, ***input-custom-mib-file***, and ***YANG-MODULE-NAME*** are variables for specific filenames:

```
$ smidump -p dependency-mib input-custom-mib-file -f yang -o YANG-MODULE-NAME.yang
```

For example:

```
$ smidump -p mib-jnx-smi.txt mib-jnx-example-custom-syslog.txt -f yang -o JUNIPER-EXAMPLE-
CUSTOM-SYSLOG-MIB.yang
```

As output, you will get the converted YANG file **JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB.yang.**

Notice that the input custom MIB file **mib-jnx-example-custom-syslog.txt** is dependent on SNMPv2-SMI, JUNIPER-SMI, and IF-MIB. But since SNMPv2-SMI and IF-MIB are standard MIBs, their definitions are already present in smidump. So, the only dependent MIB file required is **mib-jnx-smi.txt**, which has module JUNIPER-SMI definitions.

3. Copy the file **JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB.yang** to any path on the device, and copy all the dependent YANG files to the device at the following path: **/opt/lib/python2.7/site-packages/pyang/modules**.

   NOTE: You must convert all the dependent MIBs to YANG files and copy to these to the device.

   Following are some of the standard MIBs that have been converted to YANG modules and are present in the above path: **IANAifType-MIB.yang**, **ietf-yang-types.yang**, **ietf-inet-types.yang**, **IF-MIB.yang**, **JUNIPER-SMI.yang**, **SNMPv2-TC.yang**.

4. Using the CLI, load the YANG modules into snmpd using this command:

   ```
   user@host> request system yang add snmp module yang-filename package package-name
   ```

   For example:

   ```
   user@host> request system yang add snmp module /var/tmp/JUNIPER-EXAMPLE-CUSTOM-SYSLOG-
   MIB.yang package p1
   ```

   The YANG module is converted to JSON format and goes to snmpd for parsing and creating the internal database.

5. To verify the trap based on the syslog with the newly added trap definitions is working, spoof (mimic) the trap. You can do this either using the CLI or an event policy. The following is an example of spoofing the trap using the CLI.

   ```
   user@host> request snmp spoof-trap jnxExampleSyslogTrap?
   Possible completions:
   ```

```
  <trap>              The name of the trap to spoof
  jnxExampleSyslogTrap1   (Dynamic)
  jnxExampleSyslogTrap2   (Dynamic)
  jnxExampleSyslogTrap3   (Dynamic)

user@host> request snmp spoof-trap jnxExampleSyslogTrap1
Spoof-trap request result: trap sent
successfully
```

mib-jnx-example-custom-syslog.txt

```
-- ******************************************************************
-- Juniper enterprise specific custom syslog MIB.
--
-- Copyright (c) 2002-2004, 2006, Juniper Networks, Inc.
-- All rights reserved.
--
-- The contents of this document are subject to change without notice.
-- ******************************************************************

JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE, Integer32
        FROM SNMPv2-SMI
    jnxCustomMibRoot, jnxCustomSyslogNotifications
        FROM JUNIPER-SMI
    ifName
        FROM IF-MIB
;

jnxExampleCustomSyslog MODULE-IDENTITY
    LAST-UPDATED "201711270000Z"
    ORGANIZATION "Juniper Networks, Inc."
    CONTACT-INFO
            "Juniper Technical Assistance Center
             Juniper Networks, Inc.
             1133 Innovation Way
             Sunnyvale, CA 94089
             E-mail: support@juniper.net"
    DESCRIPTION
    "Example MIB objects for custom syslog"
```

```
    REVISION     "201711270000Z"
    DESCRIPTION
    "Initial draft"
    ::= { jnxCustomMibRoot 1 }

jnxExampleCustomSyslogMessage OBJECT-TYPE
        SYNTAX      OCTET STRING
        MAX-ACCESS  accessible-for-notify
        STATUS      current
        DESCRIPTION
            "The syslog message string."
        ::= { jnxExampleCustomSyslog 1 }

jnxExampleCustomSyslogInteger OBJECT-TYPE
        SYNTAX      Integer32
        MAX-ACCESS  accessible-for-notify
        STATUS      current
        DESCRIPTION
            "Example OID for adding custom Integer OID"
        ::= { jnxExampleCustomSyslog 2 }

jnxExampleSyslogTrap1 NOTIFICATION-TYPE
    OBJECTS { jnxExampleCustomSyslogMessage }
    STATUS  current
    DESCRIPTION
            "This TRAP is reserved to be sent when event 1 occurs"
    ::= { jnxCustomSyslogNotifications 1 }

jnxExampleSyslogTrap2 NOTIFICATION-TYPE
    OBJECTS { jnxExampleCustomSyslogInteger, jnxExampleCustomSyslogMessage }
    STATUS  current
    DESCRIPTION
            "This TRAP is reserved to be sent when event 2 occurs"
    ::= { jnxCustomSyslogNotifications 2 }

jnxExampleSyslogTrap3 NOTIFICATION-TYPE
    OBJECTS { ifName, jnxExampleCustomSyslogMessage }
    STATUS  current
    DESCRIPTION
            "This TRAP is reserved to be sent when event 3 occurs"
    ::= { jnxCustomSyslogNotifications 3 }
```

```
END
```

JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB.yang

```
/*
 * This YANG module has been generated by smidump 0.5.0:
 *
 *      smidump -f yang JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB
 *
 * Do not edit. Edit the source file instead!
 */

module JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB {

  namespace "urn:ietf:params:xml:ns:yang:smiv2:JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB";
  prefix "juniper-example";

  import IF-MIB {
    prefix "if-mib";
  }

  import JUNIPER-SMI {
    prefix "juniper-smi";
  }

  import ietf-yang-smiv2 {
    prefix "smiv2";
  }

  organization
   "Juniper Networks, Inc.";

  contact
   "Juniper Technical Assistance Center
    Juniper Networks, Inc.
    1133 Innovation Way
    Sunnyvale, CA 94089
    E-mail: support@juniper.net";

  description
   "Example MIB objects for custom syslog";
```

```
revision 2017-11-27 {
  description
   "Initial draft";
}


container JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB {
  config false;
}

notification jnxExampleSyslogTrap1 {
  description
   "This TRAP is reserved to be sent when event 1 occurs";
  smiv2:oid "1.3.6.1.4.1.2636.4.30.1";

  container object-1 {

    leaf jnxExampleCustomSyslogMessage {
      type binary;
      description
       "The syslog message string.";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.1";
    }
  }
}

notification jnxExampleSyslogTrap2 {
  description
   "This TRAP is reserved to be sent when event 2 occurs";
  smiv2:oid "1.3.6.1.4.1.2636.4.30.2";

  container object-1 {

    leaf jnxExampleCustomSyslogInteger {
      type int32;
      description
       "Example OID for adding custom Integer OID";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.2";
    }
  }
```

```
  container object-2 {

    leaf jnxExampleCustomSyslogMessage {
      type binary;
      description
       "The syslog message string.";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.1";
    }
  }
}

notification jnxExampleSyslogTrap3 {
  description
   "This TRAP is reserved to be sent when event 3 occurs";
  smiv2:oid "1.3.6.1.4.1.2636.4.30.3";

  container object-1 {

    leaf ifIndex {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable/if-mib:ifEntry/if-mib:ifIndex";
      }
    }

    leaf ifName {
      type leafref {
        path "/if-mib:IF-MIB/if-mib:ifTable/if-mib:ifEntry/if-mib:ifName";
      }
    }
  }

  container object-2 {

    leaf jnxExampleCustomSyslogMessage {
      type binary;
      description
       "The syslog message string.";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.1";
    }
  }
```

```
  }

  smiv2:alias "jnxExampleCustomSyslog" {
    smiv2:oid "1.3.6.1.4.1.2636.3.86.1";
  }

}
```

Limitations of Using Custom SNMP Traps

Be careful to write the event scripts in such a way that they won't trigger traps for frequently occurring syslogs. This practice avoids introducing more load on the device.

If you add an object whose access type is `readonly` or `readwrite`, that object will not be available for polling in snmp polling operations such as snmpget or snmpwalk; it will be treated as access type `notifyonly`. This is because this feature is for adding dynamic TRAP OID definitions to the device so that customer can write scripts to send custom traps for each syslog. Access types `readonly` and `readwrite` are for snmp polling, whereas `notifyonly` is for traps.

For custom MIBs, the definition of a custom table is not supported. If you want to send a trap that has a table object as a varbind, use the already defined table in Junos MIBs rather than defining a custom table in your custom MIB.

The YANG file needs to be loaded on all the chassis nodes and Routing Engines separately. The `request system yang add` command does not automatically copy it to backup Routing Engine.

Example Custom Syslog Trap

This example custom syslog trap illustrates a use case in which the operator wants to receive traps when either of the following occur:

- A user enters the configuration mode in the CLI (event defined as `ui_dbase_login_event`)
- A user does a commit (event defined as `ui_commit`)

Before the custom syslog trap feature was supported, the only way to do this was to use jnxSyslogTrap, which has a fixed OID, for both events. With the custom syslog trap feature, you can now generate traps that have custom defined OIDs.

To define a custom syslog trap:

1. Use the sample file provided and convert it to **JUNIPER-EXAMPLE-CUSTOM-SYSLOIG-MIB.yang** .

```
smidump -p mib-jnx-smi.txt mib-jnx-example-custom-syslog.txt -f yang -o JUNIPER-EXAMPLE-
CUSTOM-SYSLOG-MIB.yang
```

2. Copy the YANG file onto your device.
3. Load the SNMP YANG file.

```
root@host> request system yang add snmp package p1 module ~/JUNIPER-EXAMPLE-CUSTOM-SYSLOG-
MIB.yang
```

4. Copy the slax script to **/var/db/scripts/event** to spoof the trap .

   For ui_dbase_login_event, you will configure the enteredConfigMode trap which has the username varbind.

   For ui_commit, you will configure the configCommitted trap which has the username command an comment as three varbinds.
5. Configure the trap:

```
set event-options policy custom-trap events ui_dbase_login_event
set event-options policy custom-trap events ui_commit
set event-options policy custom-trap then event-script custom-trap.slax
set event-options event-script file custom-trap.slax
```

6. Enable snmpd traceoptions and trap target to verify the traps that are sent.

```
set snmp trap-group trap-group targets ip-address
set snmp traceoptions flag all
```

7. Verify trap is working.

Sample MIB file

```
-- ****************************************************************
-- Juniper enterprise specific custom syslog MIB.
--
-- Copyright (c) 2002-2004, 2006, Juniper Networks, Inc.
-- All rights reserved.
--
-- The contents of this document are subject to change without notice.
```

```
-- ****************************************************************

JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE
        FROM SNMPv2-SMI
    jnxCustomMibRoot, jnxCustomSyslogNotifications
        FROM JUNIPER-SMI
;

jnxExampleCustomSyslog MODULE-IDENTITY
    LAST-UPDATED "201806220000Z"
    ORGANIZATION "Juniper Networks, Inc."
    CONTACT-INFO
            "Juniper Technical Assistance Center
             Juniper Networks, Inc.
             1133 Innovation Way
             Sunnyvale, CA 94089
             E-mail: support@juniper.net"
    DESCRIPTION
    "Example MIB objects for custom syslog"
    REVISION     "201806220000Z"
    DESCRIPTION
    "Initial draft"
    ::= { jnxCustomMibRoot 1 }

username OBJECT-TYPE
        SYNTAX      OCTET STRING
        MAX-ACCESS  accessible-for-notify
        STATUS      current
        DESCRIPTION
            "Username"
        ::= { jnxExampleCustomSyslog 1 }

command OBJECT-TYPE
        SYNTAX      OCTET STRING
        MAX-ACCESS  accessible-for-notify
        STATUS      current
        DESCRIPTION
            "Executed command"
        ::= { jnxExampleCustomSyslog 2 }
```

```
comment OBJECT-TYPE
        SYNTAX      OCTET STRING
        MAX-ACCESS  accessible-for-notify
        STATUS      current
        DESCRIPTION
            "Additional comment"
        ::= { jnxExampleCustomSyslog 3 }

enteredConfigMode NOTIFICATION-TYPE
    OBJECTS { username }
    STATUS  current
    DESCRIPTION
            "This TRAP is sent when a user enteres config mode. "
    ::= { jnxCustomSyslogNotifications 1 }

configCommitted NOTIFICATION-TYPE
    OBJECTS { username, command, comment }
    STATUS  current
    DESCRIPTION
            "This TRAP is sent when a user does config commit"
    ::= { jnxCustomSyslogNotifications 2 }

END
```

Sample YANG Converted File

```
/*
 * This YANG module has been generated by smidump 0.5.0:
 *
 *      smidump -f yang JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB
 *
 * Do not edit. Edit the source file instead!
 */

module JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB {

  namespace "urn:ietf:params:xml:ns:yang:smiv2:JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB";
  prefix "juniper-example";

  import JUNIPER-SMI {
    prefix "juniper-smi";
  }
```

```
import ietf-yang-smiv2 {
  prefix "smiv2";
}

organization
 "Juniper Networks, Inc.";

contact
 "Juniper Technical Assistance Center
  Juniper Networks, Inc.
  1133 Innovation Way
  Sunnyvale, CA 94089
  E-mail: support@juniper.net";

description
 "Example MIB objects for custom syslog";

revision 2018-06-22 {
  description
   "Initial draft";
}


container JUNIPER-EXAMPLE-CUSTOM-SYSLOG-MIB {
  config false;
}

notification enteredConfigMode {
  description
   "This TRAP is sent when a user enteres config mode. ";
  smiv2:oid "1.3.6.1.4.1.2636.4.30.1";

  container object-1 {

    leaf username {
      type binary;
      description
       "Username";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.1";
    }
  }
```

```
}

notification configCommitted {
  description
   "This TRAP is sent when a user does config commit";
  smiv2:oid "1.3.6.1.4.1.2636.4.30.2";

  container object-1 {

    leaf username {
      type binary;
      description
       "Username";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.1";
    }
  }

  container object-2 {

    leaf command {
      type binary;
      description
       "Executed command";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.2";
    }
  }

  container object-3 {

    leaf comment {
      type binary;
      description
       "Additional comment";
      smiv2:max-access "accessible-for-notify";
      smiv2:oid "1.3.6.1.4.1.2636.3.86.1.3";
    }
  }
}

smiv2:alias "jnxExampleCustomSyslog" {
  smiv2:oid "1.3.6.1.4.1.2636.3.86.1";
```

```
  }

}
```

slax Script cutom_trap.slax (in /var/db/scripts/event)

```
version 1.0;
ns junos = "http://xml.juniper.net/junos/*/junos";
ns xnm = "http://xml.juniper.net/xnm/1.1/xnm";
ns jcs = "http://xml.juniper.net/junos/commit-scripts/1.0";
import "../import/junos.xsl";
match / {
    <event-script-results> {
        expr jcs:syslog("external.warning",event-script-input/trigger-event/id);
        var $id = event-script-input/trigger-event/id;
        if ($id == 'UI_DBASE_LOGIN_EVENT'){
            var $committing-user = event-script-input/trigger-event/attribute-list/
attribute[name=="username"]/value;
            var $requestSnmpTrap = <request-snmp-spoof-trap> {
                <trap> "enteredConfigMode";
                <variable-bindings>
                    "username=" _ $committing-user;
            }
            var $snmpTrapResults = jcs:invoke( $requestSnmpTrap );
        }
        else if ($id == 'UI_COMMIT'){
           var $committing-user = event-script-input/trigger-event/attribute-list/
attribute[name=="username"]/value;
           var $committing-command = event-script-input/trigger-event/attribute-list/
attribute[name=="command"]/value;
           var $committing-comment = event-script-input/trigger-event/attribute-list/
attribute[name=="message"]/value;

            var $requestSnmpTrap = <request-snmp-spoof-trap> {
                <trap> "configCommitted";
                <variable-bindings>
                    "username=" _ $committing-user _ ", command=" _ $committing-command _ ",
comment=" _ $committing-comment;
            }
            var $snmpTrapResults = jcs:invoke( $requestSnmpTrap );
        }
```

```
    }
}
```

Release History Table

Release	Description
20.1R1	Starting in Junos OS Release 20.1, after graceful routing engine switchover (GRES), the new primary Routing Engine sends a single warmStart notification.

Trace SNMP Activity

IN THIS SECTION

- Monitor SNMP Activity and Track Problems That Affect SNMP Performance on a Device Running Junos OS | **514**
- Trace SNMP Activity on a Device Running Junos OS | **518**
- Example: Tracing SNMP Activity | **522**
- Configure the Certificate Expiration Trap | **523**
- Enable Peer Down and IPsec Tunnel Down Traps | **524**

Monitor SNMP Activity and Track Problems That Affect SNMP Performance on a Device Running Junos OS

IN THIS SECTION

- Check for MIB Objects Registered with SNMPd | **515**
- Track SNMP Activity | **516**
- Monitor SNMP Statistics | **516**
- Check CPU Utilization | **517**
- Check Kernel and Packet Forwarding Engine Response | **517**

On Junos OS devices, you can view the information about monitoring the SNMP activity and identifying the problems that impact the SNMP performance:

Check for MIB Objects Registered with SNMPd

To access data related to a MIB object, the MIB object must be registered with the snmpd. When an SNMP subagent is online, it registers the associated MIB objects with the snmpd. The snmpd maintains a mapping of the objects and the subagents with which the objects are associated. However, the registration attempt fails occasionally, and the objects remain unregistered with the snmpd until the next time the subagent restarts and successfully registers the objects.

When a network management system polls for data related to objects that are not registered with the snmpd, the snmpd returns either a `noSuchName` error (for SNMPv1 objects) or a `noSuchObject` error (for SNMPv2 objects).

You can use the following commands to check for MIB objects that are registered with the snmpd:

- `show snmp registered-objects`—Creates a `/var/log/snmp_reg_objs` file that contains the list of registered objects and their mapping to various subagents.
- `file show /var/log/snmp_reg_objs`—Displays the contents of the `/var/log/snmp_reg_objs` file.

The following example shows the steps for creating and displaying the `/var/log/snmp_reg_objs` file:

```
user@host> show snmp registered-objects
user@host> file show /var/log/snmp_reg_objs
------------------------------------------------------------
 Registered MIB Objects
 root_name =
------------------------------------------------------------
.1.2.840.10006.300.43.1.1.1.1.2 (dot3adAggMACAddress) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.3 (dot3adAggActorSystemPriority) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.4 (dot3adAggActorSystemID) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.5 (dot3adAggAggregateOrIndividual) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.6 (dot3adAggActorAdminKey) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.7 (dot3adAggActorOperKey) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.8 (dot3adAggPartnerSystemID) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.9 (dot3adAggPartnerSystemPriority) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.10 (dot3adAggPartnerOperKey) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.1.1.11 (dot3adAggCollectorMaxDelay) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.1.2.1.1 (dot3adAggPortListPorts) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.2 (dot3adAggPortActorSystemPriority) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.3 (dot3adAggPortActorSystemID) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.4 (dot3adAggPortActorAdminKey) (/var/run/mib2d-11)
```

```
.1.2.840.10006.300.43.1.2.1.1.5 (dot3adAggPortActorOperKey) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.6 (dot3adAggPortPartnerAdminSystemPriority) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.7 (dot3adAggPortPartnerOperSystemPriority) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.8 (dot3adAggPortPartnerAdminSystemID) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.9 (dot3adAggPortPartnerOperSystemID) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.10 (dot3adAggPortPartnerAdminKey) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.11 (dot3adAggPortPartnerOperKey) (/var/run/mib2d-11)
.1.2.840.10006.300.43.1.2.1.1.12 (dot3adAggPortSelectedAggID) (/var/run/mib2d-11)
---(more)---
```

The **/var/log/snmp_reg_objs** file contains objects that are associated with the Junos OS processes which are registered with the snmpd. You can view the objects using the `show snmp registered-objects` command. If a MIB object related to a Junos OS process that is up and running is not shown in the list of registered objects, you might want to restart the software process to retry object registration with the snmpd.

Track SNMP Activity

SNMP tracing operations track activity of SNMP agents and record the information in log files. By default, Junos OS does not trace any SNMP activity. To enable tracking of SNMP activities on a device running Junos OS, include the `set traceoptions flag all` statement at the `[edit snmp]` hierarchy level.

The following log files are created:

- snmpd
- mib2d
- rmopd

You can use the `show log` *`log-filename`* operational command to view the contents of the log file. In the snmpd log file (see the following example), a sequence of >>> represents an incoming packet, whereas a sequence of <<< represents an outgoing packet. You can use the source and request ID combinations to match requests and responses, if there are multiple network management systems polling the device at the same time. Response log is not created in the log file if the SNMP master agent or the SNMP subagent has not responded to a request.

You can analyze the request-response time to identify and understand delayed responses.

You can review the log file using the `show log snmpd` command.

Monitor SNMP Statistics

The `show snmp statistics extensive` operational command provides you with an option to review SNMP traffic, including traps, on a device. Output for the `show snmp statistics extensive` command shows real-time values and can be used to monitor values such as throttle drops, currently active, max active, not

found, time out, max latency, current queued, total queued, and overflows. You can identify slowness in SNMP responses by monitoring the currently active count, because a constant increase in the currently active count is directly linked to slow or no response to SNMP requests.

Check CPU Utilization

High CPU usage of the software processes that are being queried, such as snmpd or mib2d, is another factor that can lead to slow response or no response. You can use the `show system processes extensive` operational command to check the CPU usage levels of the Junos OS processes.

Check Kernel and Packet Forwarding Engine Response

As mentioned in "Understand SNMP Implementation in Junos OS" on page 371, some SNMP MIB data are maintained by the kernel or Packet Forwarding Engine. For such data to be available for the network management system, the kernel has to provide the required information to the SNMP subagent in mib2d. A slow response from the kernel can cause a delay in mib2d returning the data to the network management system. Junos OS adds an entry in the mib2d log file every time that an interface takes more than 10,000 microseconds to respond to a request for interface statistics. You can use the `show log` *`log-filename`* `| grep “kernel response time”` command to find out the response time taken by the kernel.

Checking the Kernel Response Time

```
user@host> show log mib2d | grep “kernel response time”
 Aug 17 22:39:37  == kernel response time for
 COS_IPVPN_DEFAULT_OUTPUT-t1-7/3/0:10:27.0-o: 9.126471 sec, range
 (0.000007, 11.000806)

 Aug 17 22:39:53  == kernel response time for
 COS_IPVPN_DEFAULT_INPUT-t1-7/2/0:5:15.0-i: 5.387321 sec, range
 (0.000007, 11.000806)

 Aug 17 22:39:53  == kernel response time for ct1-6/1/0:9:15: 0.695406
 sec, range (0.000007, 11.000806)

 Aug 17 22:40:04  == kernel response time for t1-6/3/0:6:19: 1.878542
 sec, range (0.000007, 11.000806)

 Aug 17 22:40:22  == kernel response time for lsq-7/0/0: 2.556592 sec,
 range (0.000007, 11.000806)
```

Trace SNMP Activity on a Device Running Junos OS

IN THIS SECTION

- Configure the Number and Size of SNMP Log Files | 519
- Configure Access to the Log File | 519
- Configure a Regular Expression for Lines to Be Logged | 520
- Configure the Trace Operations | 520

SNMP tracing operations track activity for SNMP agents and record the information in log files. The logged error descriptions provide detailed information to solve problems.

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 Explorer.)
- 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:

Configure the Number and Size of SNMP Log Files

By default, when the trace file reach 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).

Configure 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;
```

Configure 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;
```

Configure 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 46 on page 521 describes the meaning of the SNMP tracing flags.

Table 46: 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
`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

Table 46: SNMP Tracing Flags *(Continued)*

Flag	Description	Default Setting
`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.

Example: Tracing SNMP Activity

Trace information about SNMP packets:

```
[edit]
snmp {
    traceoptions {
        file size 10k files 5;
        flag pdu;
        flag protocol-timeouts;
        flag varbind-error;
    }
}
```

Configure the Certificate Expiration Trap

Before you begin:

- Understand how certificates works on VPN. Read Understanding Certificate Chains.

This topic shows how to configure certificate expiration trap and configure the number of days before to generate the trap.

1. Configure the number of days before to generate the trap for all certificates.

```
user@host# set security pki trap all-certificates number-of-days
```

2. Configure the number of days before to generate the trap for CA certificate.

```
user@host# set security pki trap ca-identity ca-profile-name number-of-days
```

3. Configure the number of days before to generate the trap for local certificate.

```
user@host# set security pki trap certificate-idcertificate-id-name number-of-days
```

4. Confirm your configuration by entering the `show security pki trap` command.

```
user@host# show security pki trap
certificate-id crt_spk1 {
    30;
}
ca-identity Root-CA {
    30;
}
all-certificates {
    30;
}
```

SEE ALSO

trap (Security PKI)

show security ipsec statistics

Enable Peer Down and IPsec Tunnel Down Traps

This topic shows how to enable `peer-down` and `ipsec-tunnel-down` traps.

1. Enable the IKE trap peer down. Trap gets generated when the peer is down.

```
user@host# set security ike trap peer-down
```

2. Enable the IKE trap IPsec tunnel down. Trap gets generated when the peer is up and the IPsec SA is down.

```
user@host# set security ike trap ipsec-tunnel-down
```

3. Confirm your configuration by entering the `show security ike trap` command.

```
user@host# show security ike trap
ipsec-tunnel-down;
peer-down;
```

SEE ALSO

trap (Security PKI)

show security ipsec statistics

Access Privileges for an SNMP Group

IN THIS SECTION

- Configure the Access Privileges Granted to a Group | 525
- Example: Configure the Access Privileges Granted to a Group | 529
- Assign Security Model and Security Name to a Group | 530
- Example: Security Group Configuration | 532

SNMP version 3 (SNMPv3) uses the view-based access control model (VACM), which allows you to configure the access privileges granted to a group. You can control the access 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 a particular security level (authenticated, privacy, or none). For information about how to configure views, see "Configure MIB Views" on page 573.

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. You can define the access 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. For more information about this statement, see *vacm*.

Configure the Access Privileges Granted to a Group

IN THIS SECTION

- Configure the Group | **526**
- Configure the Security Model | **526**
- Configure the Security Level | **527**

- Associate MIB Views with an SNMP User Group | 527

This topic includes the following sections:

Configure 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.

Configure 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

Configure 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.

You can grant access privileges 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.

Associate MIB Views with an SNMP User Group

IN THIS SECTION

- Configure the Notify View | 528
- Configure the Read View | 528
- Configure the Write View | 528

MIB views define access privileges for members of a group. You can apply separate views 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. For more information about this statement, see *access (SNMP)*.

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 "Configure MIB Views" on page 573.

This section describes the following topics related to this configuration:

Configure 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. For more information about this statement, see *notify-view*.

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.

Configure 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. For more information about this statement, see *read-view*.

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

Configure 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. For more information about this statement, see *write-view*.

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

Example: Configure the Access Privileges Granted to a Group

Define access privileges:

```
[edit snmp v3 vacm]
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;
                }
            }
        }
    }
}
```

Assign Security Model and Security Name to a Group

IN THIS SECTION

- Configure the Security Model | 530
- Assign Security Names to Groups | 531
- Configure the Group | 531

To assign security names to groups, include the following statements at the `[edit snmp v3 vacm security-to-group]` hierarchy level. For more information about this statement, see *security-model (Group)*.

This topic includes the following sections:

Configure 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

Assign 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 " Create SNMPv3 Users" on page 535. For information about configuring a community string, see " Configure SNMPv3 Community" on page 566.

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.

Configure 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 defines what SNMP objects can read, writte to, or create. A group also defines the notifications a user can receive.

If you already have a group that is configured with all 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. For more information about this statement, see *group (Defining Access Privileges for an SNMPv3 Group)*.

Example: Security Group Configuration

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;
            }
        }
    }
}
```

Configure Local Engine ID on SNMPv3

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. For more information about this statement, see *engine-id*.

NOTE: If you are using SNMPv3 and if the engine ID is based on the MAC address and you upgrade from an earlier release to one of the releases (14.1X53-D50, 16.1R5, 17.1R2, 17.2R1, 15.1X53-D231, 14.1X53-D43, 15.1X53-D232), you must reconfigure SNMPv3 because the engine ID is changed by the upgrade. If you do not reconfigure SNMPv3, you will see authentication error for SNMPv3 polling because the engine ID is changed after the upgrade. You only need to reconfigure SNMPv3 on the first such upgrade. If you then upgrade from one of the mentioned releases to another of these releases, you do not have to upgrade SNMPv3 again.

To reconfigure SNMPv3, use the following procedure. Do not use the `rollback 1` command.

1. Check what the SNMPv3 configuration is.

```
user@host# show snmp v3
```

2. Delete the SNMPv3 configuration.

```
user@host# delete snmp v3
```

3. Reconfigure SNMPv3 configuration (see ouput from Step 1).

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 primary IP address of the device if the device has multiple routing engines and has the primary IP address configured. Alternatively, you can use the MAC address of the management port if the device has only one Routing Engine.

Configure SNMPv3

IN THIS SECTION

- Create SNMPv3 Users | **535**
- Minimum SNMPv3 Configuration on a Device Running Junos OS | **535**
- Example: SNMPv3 Configuration | **536**

The QFX3500 switch supports SNMP version 3 (SNMPv3). SNMPv3 enhances the functionality of SNMPv1 and SNMPv2c by supporting user authentication and data encryption. SNMPv3 uses the user-based security model (USM) to provide security for SNMP messages, and the view-based access control model (VACM) for user access control.

SNMPv3 features include:

- With USM, the SNMP messages between the SNMP manager and the agent can have the message source authenticated and the data integrity checked. USM reduces messaging delays and message replays by enforcing timeout limits and by checking for duplicate message request IDs.
- VACM complements USM by providing user access control for SNMP queries to the agent. You define access privileges that you wish to extend to a group of one or more users. Access privileges are determined by the security model parameters (`usm`, `v1`, or `v2`) and security level parameters (`authentication`, `privacy`, or `none`). For each security level, you must associate one MIB view for the group. Associating a MIB view with a group grants the read, write, or notify permission to a set of MIB objects for the group.
- You configure security parameters for each user, including the username, authentication type and authentication password, and privacy type and privacy password. The username given to each user is in a format that is dependent on the security model configured for that user.
- To ensure messaging security, another type of username, called the security name, is included in the messaging data that is sent between the local SNMP server and the destination SNMP server. Each user name is mapped to a security name, but the security name is in a format that is independent of the security model.
- 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 that 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 the address of an SNMP management application and other attributes used in sending notifications. Target parameters define the message processing and security parameters used in sending notifications to a particular target.

To configure SNMPv3, perform the following tasks:

- "Configure MIB Views" on page 573
- "Access Privileges for an SNMP Group" on page 524
- "Configure SNMPv3 Traps on a Device Running Junos OS" on page 543
- "Configure SNMP Informs" on page 550

Create SNMPv3 Users

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 written to the configuration file. After the generation of the key, you can delete the password from this configuration file.

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.

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

Minimum SNMPv3 Configuration on a Device Running Junos OS

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

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

1. Create users and configure authentication.

   `user@host#` **set snmp v3 usm local-engine user superuser authentication-md5 authentication-password 12345678**

   `user@host#`**set snmp v3 usm local-engine user superuser privacy-aes128 privacy-password 12345678**

2. Configure access privileges to a group.

   `user@host#` **set snmp v3 vacm access group supergroup default-context-prefix security-model usm security-level authentication context-match exact**

   `user@host#` **set snmp v3 vacm access group supergroup default-context-prefix security-model usm security-level authentication read-view readview**

   `user@host#` **set snmp v3 vacm access group supergroup default-context-prefix security-model usm security-level authentication write-view writeview**

   `user@host#` **set snmp v3 vacm access group supergroup default-context-prefix security-model usm security-level authentication notify-view notifyview**

   `user@host#` **set snmp v3 vacm security-to-group security-model usm security-name superuser group supergroup**

3. (Optional) Configure the target address properties to which the trap notification is sent.

   `user@host#` **set snmp v3 target-address TA address <*nms-ipaddress*> tag-list trap_recv target-parameters tp1**

```
user@host# set snmp v3 target-parameters tp1 parameters message-processing-model v3 security-
model usm security-level authentication security-name superuser
user@host# set snmp v3 target-parameters tp1 notify-filter nfilter1
user@host# set snmp v3 notify-filter nfilter1 oid .1 include
user@host# set snmp v3 notify notify1 type trap tag trap_recv
```

4. Configure snmp view to read, write and notify access to the MIB.

```
user@host# set snmp view readview oid .1 include
user@host# set snmp view writeview oid .1 include
user@host# set snmp view notifyview oid .1 include
```

SEE ALSO

v3

Example: SNMPv3 Configuration

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 john { # security-name john is defined here
            authentication-md5 {
                authentication-password authentication-password;
            }
            privacy-des {
                privacy-password privacy-password;
            }
        }
        user bob { # security-name bob is defined here
            authentication-sha {
                authentication-password authentication-password;
            }
            privacy-none;
        }
        user julia { # security-name julia is defined here
            authentication-none;
            privacy-none;
        }
```

```
        user lauren { # security-name lauren is defined here
            authentication-sha {
                authentication-password authentication-password;
            }
            privacy-aes128 {
                privacy-password privacy-password;
            }
        }
        user richard { # security-name richard is defined here
            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 security group san-fancisco
                group san-francisco;
            }
            security-name bob { # Assigns bob to security group new-york
                group new-york;
            }
            security-name julia {# Assigns julia to security group chicago
                group chicago;
            }
            security-name lauren {# Assigns lauren to security group paris
                group paris;
```

```
            }
            security-name richard {# Assigns richard to security group geneva
                group geneva;
            }
        }
    }
}
```

Configure SNMPv3 Authentication Type and Encryption Type

IN THIS SECTION

- Configure SNMPv3 Authentication Type | 540
- Configure SNMPv3 Encryption Type | 542

Configure SNMPv3 Authentication Type

IN THIS SECTION

- Configur MD5 Authentication | 541
- Configure SHA Authentication | 541
- Configure No Authentication | 541

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

This topic includes the following sections:

Configur 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. For more information about this statement, see *authentication-md5*.

Configure SHA Authentication

You can configure the following secure hash algorithm (SHA) as the authentication type for an SNMPv3 user:

- `authentication-sha`
- `authentication-sha224`
- `authentication-sha256`

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. For more information about this statement, see *authentication-sha*.

To configure the secure hash algorithm (SHA) as the authentication type for an SNMPv3 user, include the `authentication-sha224` at the `[edit snmp v3 usm local-engine user username]` hierarchy level. For more information about this statement, see *authentication-sha224*.

To configure the secure hash algorithm (SHA) as the authentication type for an SNMPv3 user, include the `authentication-sha256` statement at the `[edit snmp v3 usm local-engine user username]` hierarchy level. For more information about this statement, see *authentication-sha256*.

Configure 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. For more information about this statement, see *authentication-none*.

SEE ALSO

v3

Configure SNMPv3 Encryption Type

IN THIS SECTION

- Configure Advanced Encryption Standard Algorithm | 542
- Configure Data Encryption Algorithm | 542
- Configure Triple DES | 542
- Configure No Encryption | 542

By default, encryption is set to none.

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

This topic includes the following sections:

Configure 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. For more information about this statement, see *privacy-aes128*.

Configure 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. For more information about this statement, see *privacy-des*.

Configure 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. For more information about this statement, see *privacy-3des*.

Configure 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. For more information about this statement, see *privacy-none*.

SNMPv3 Traps

IN THIS SECTION

- Configure SNMPv3 Traps on a Device Running Junos OS | 543
- Configure SNMPv3 Trap Notification | 544
- Example: Configure SNMPv3 Trap Notification | 544
- Configure the Trap Notification Filter | 545
- Configure the Trap Target Address | 545
- Example: Configure the Tag List | 547
- Define and Configure the Trap Target Parameters | 548

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.

Configure SNMPv3 Traps on a Device Running Junos OS

The target address defines a management application's address and parameters used in sending notifications. Target parameters define the message processing and security parameters used in sending notifications to a particular management target. SNMPv3 also lets you define SNMPv1 and SNMPv2c traps.

NOTE: When you configure SNMP traps, ensure your configured access privileges allow the traps to be sent. You can configure access privileges at the `[edit snmp v3 vacm access]` and `[edit snmp v3 vacm security-to-group]` hierarchy levels.

For details on SNMP v1 or v2 trap to OID translation and trap details that are sent by each category, see MIB Explorer.

Configure SNMPv3 Trap Notification

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.

Each notify entry name must be unique.

Junos OS supports two types of notification: `trap` and `inform`.

SEE ALSO

v3

Example: Configure SNMPv3 Trap Notification

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;
}
```

Configure the Trap Notification Filter

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. You can represent the subtree 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.

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. For more information about this statement, see *notify-filter (Configuring the Profile Name)*.

Configure the Trap Target Address

IN THIS SECTION

- Configure the Address | **546**
- Configure the Address Mask | **546**
- Configure the Port | **546**
- Configure the Routing Instance | **547**
- Configure the Trap Target Address | **547**
- Apply Target Parameters | **547**

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.

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.

To configure the target address properties, include the following statements at the `[edit snmp v3 target-address` *`target-address-name`*`]` hierarchy level:

Unlike with SNMP v2, In SNMPv3, there is no configuration option to limit inbound polling. But you can configure a lo0 filter to limit inbound polling by creating a rule to allow SNMP from your monitoring system IPs. For example:

```
set policy-options prefix-list SNMP 10.1.1.1/32
set policy-options prefix-list SNMP 192.168.1.0/24

set firewall family inet filter CoPP term SNMP from source-prefix-list SNMP
set firewall family inet filter CoPP term SNMP from protocol udp
set firewall family inet filter CoPP term SNMP from destination-port snmp
set firewall family inet filter CoPP term SNMP then accept
set firewall family inet filter CoPP term SNMP then count SNMP
```

Configure the Address

To configure the address, include the `address` statement at the `[edit snmp v3 target-address` *`target-address-name`*`]` hierarchy level. For more information about this statement, see *address*.

`address` is the SNMP target address.

Configure 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. *address-mask*.

`address-mask` combined with the address defines a range of addresses.

Configure 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. For more information about this statement, see *port*.

Configure 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. For more information about this statement, see *routing-instance*.

Configure 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. For more information about this statement, see *tag-list*.

`tag-list` specifies one or more tags as a space-separated list enclosed within double quotes.

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.

Apply 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:

Example: Configure the Tag List

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;
}
```

Define and Configure the Trap Target Parameters

IN THIS SECTION

- Apply the Trap Notification Filter | 549
- Configure the Target Parameters | 549

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:

For more information about configuring subscriber secure policies, see *Subscriber Secure Policy Overview*.

This topic includes the following sections:

Apply 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. For more information about this statement, see *notify-filter (Applying to the Management Target)*.

Configure the Target Parameters

IN THIS SECTION

- Configure the Message Processing Model | **549**
- Configure the Security Model | **549**
- Configure the Security Level | **550**
- Configure the Security Name | **550**

To configure target parameter properties, include the following statements at the `[edit snmp v3 target-parameters target-parameter-name parameters]` hierarchy level.

This section includes the following topics:

Configure 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. For more information about this statement, see *message-processing-model*.

The subscriber secure policy on MX Series routers requires the `v3` message-processing model. See *Subscriber Secure Policy Overview*.

Configure 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. For more information about this statement, see *security-model (SNMP Notifications)*.

The subscriber secure policy on MX Series routers requires the `usm` security model. See *Subscriber Secure Policy Overview*.

Configure 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. For more information about this statement, see *security-level (Generating SNMP Notifications)*.

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.

The subscriber secure policy on MX Series routers requires the `privacy` security level . See *Subscriber Secure Policy Overview* for more information.

Configure 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. For more information about this statement, see *security-name (SNMP Notifications)*.

If you use USM as security model, the `security-name` identifies the user that is used when the notification is generated. If you use v1 or v2c as security models, `security-name` identifies the SNMP community used when the notification is generated.

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.

SNMPv3 Informs

IN THIS SECTION

- Example: Configure the Inform Notification Type and Target Address | 552

Example: Configure the Remote Engine ID and Remote User | 553

Junos OS supports two types of notifications: traps and informs.

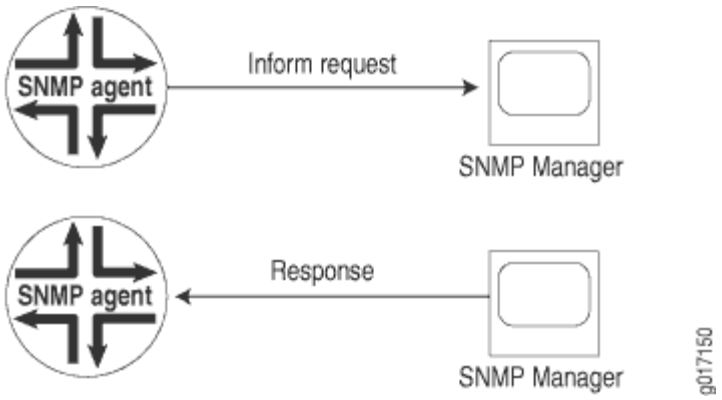
With traps, the receiver does not send any acknowledgment when it receives a trap. Therefore, the sender cannot determine if the trap was received. A trap may be lost because a problem occurred during transmission. To increase reliability, an inform is similar to a trap except that the inform is stored 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. 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 23: Inform Request and Response



Example: Configure the Inform Notification Type and Target Address

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;
}
```

Example: Configure the Remote Engine ID and Remote User

IN THIS SECTION

- Requirements | 553
- Overview | 553
- Configuration | 555
- Verification | 556

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

This feature requires the use of plain-text passwords valid for SNMPv3. SNMPv3 has the following requirements when you create plain-text passwords on a router or a 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.

It is best to use quotation marks to enclose passwords although it is not necessary. You need quotation marks if the password contains any spaces or 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 47 on page 554 to use in configuring the remote engine ID and remote user in this example.

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).

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.

Table 47: 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

IN THIS SECTION

- CLI Quick Configuration | 555
- Configuring the Remote Engine and Remote User | 555
- Results | 556

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.

```
set usm remote-engine 800007E5804089071BC6D10A41 user u10 authentication-md5 authentication-
password "qol67R%?"
set usm remote-engine 800007E5804089071BC6D10A41 user u10 privacy-des privacy-password "m*72Jl9v"
```

Configuring the Remote Engine and Remote User

Step-by-Step Procedure

The following example requires that you navigate to various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide*.

To configure the remote engine ID and remote user:

1. 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-password "qol67R%?"
```

2. 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-
password "m*72Jl9v"
```

Results

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 "$9$hagSyKNdbY2acyvLN-2g69CtpBRhSvMX/CLx-
V4oZUjkqfQz69CuF36Apu1Idbw2ZUiHm3/C.mF/CA1IVws4oGkqf6CtzF";## SECRET-DATA
            }
            privacy-des {
                privacy-key "$9$GJDmf3nCtO1zFnCu0hcrevM87bs2oaUbwqmP5F3Ap0O1hrevMLxcSYgoaUDqmf5n/
Ap0REyk.BIREyr4aJZUHfTz9tu5T";## SECRET-DATA
            }
        }
    }
}
```

After you have confirmed that the configuration is correct, enter `commit` from configuration mode.

Verification

IN THIS SECTION

- Verifying the Configuration of the Remote Engine ID and Username | 557

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

SEE ALSO

show snmp v3

SNMP Communities

IN THIS SECTION

- Configure SNMP Communities | 558
- Configure SNMP Community String | 564
- Examples: Configure the SNMP Community String | 564
- Configure the SNMPv3 Community | 566
- Example: Configure SNMPv3 Community | 568

An SNMP community defines the level of authorization granted to its members, such as the available MIB objects, the operations (read-only or read-write) that are valid for those objects, and the authorized SNMP clients, based on their source IP addresses.

Configure SNMP Communities

IN THIS SECTION

- Add a Group of Clients to an SNMP Community | 562

Configuring the SNMP agent in Junos OS is a straightforward task that shares familiar settings with 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.

The SNMP community string defines the relationship between an SNMP server system and the client system. This string is a password to control the client's access to the server.

To create a read-only SNMP community:

1. Enter the SNMP community used in your network.

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

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.

```
[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
```

2. 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 "Configure MIB Views" on page 573.

   ```
   [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
   ```

3. 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. List the clients by IP address and prefix. Typically, the list includes the SNMP network management system in your network or the address of your management network. If no

clients statement is present, all clients are allowed. For *address*, you must specify an IPv4 or IPv6 address, not a hostname.

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

The following statement defines the hosts in the 192.168.1.0/24 network as being authorized in the public community.

```
[edit groups global snmp community public]
user@host# set clients 192.168.1.0/24
```

4. Define the clients that are not authorized within the community by specifying their IP address, followed by the restrict statement.

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

The following statement defines all other hosts as being restricted from the public community.

```
[edit groups global snmp community public]
user@host# set clients 0/0 restrict
```

5. At the top level of the configuration, apply the configuration group.

   If you use a configuration group, you must apply it for it to take effect.

```
[edit]
user@host# set apply-groups global
```

6. Commit the configuration.

```
user@host# commit
```

To create a read-write SNMP community:

1. Enter the SNMP community used in your network.

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

This example standard community string `private` to identify the community granted read-write access to the SNMP agent running on the device.

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

2. Define the authorization level for the community.

```
[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-write
```

3. Define a list of clients in the community who are authorized to make changes to the SNMP agent in Junos OS.

List the clients by IP address and prefix.

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

For example:

```
[edit groups global snmp community private]
user@host# set clients 192.168.1.15/24
user@host# set clients 192.168.1.18/24
```

4. Define the clients that are not authorized within the community by specifying their IP address, followed by the `restrict` statement.

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

   The following statement defines all other hosts as being restricted from the public community.

```
[edit groups global snmp community private]
user@host# set clients 0/0 restrict
```

5. At the top level of the configuration, apply the configuration group.

   If you use a configuration group, you must apply it for it to take effect.

```
[edit]
user@host# set apply-groups global
```

6. Commit the configuration.

```
user@host# commit
```

Add a Group of Clients to an SNMP Community

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, use the `set snmp client-list` *client-list-name* statement followed by the IP addresses of the clients.

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 Policies, Firewall Filters, and Traffic Policers User Guide.

To add a client list or prefix list to an SNMP community, use the `set snmp commmunity` *community-name* `client-list-name` statement.

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;
    }
}
```

Configure SNMP Community String

The SNMP community string defines the relationship between an SNMP server system and the client system. This string acts like a password to control the client's access to the server.

To configure a community string in a Junos OS configuration, use the `set snmp community` statement.

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

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 "Configure MIB Views" on page 573.

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

Community names must be unique within each SNMP system.

SEE ALSO

> *community*

Examples: Configure the SNMP Community String

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.
        }
    }
}
```

Configure the SNMPv3 Community

IN THIS SECTION

- Configuring the Community Name | 567
- Configuring the Context | 567
- Configuring the Security Names | 568
- Configuring the Tag | 568

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-name community-name;
context context-name;
security-name security-name;
tag tag-name;
```

The following is a minimal set of sample configuration that is needed for `snmp v3 snmp-community` configuration:

```
set snmp v3 vacm security-to-group security-model v2c security-name NOSNMPV3 group SNMPV3GROUP
set snmp v3 vacm access group SNMPV3GROUP default-context-prefix security-model any security-
level none read-view SNMPVIEW
set snmp v3 vacm access group SNMPV3GROUP default-context-prefix security-model any security-
```

```
level none write-view SNMPVIEW
set snmp v3 snmp-community SNMPV3COMMUNITY community-name JTACCOMMUNITY
set snmp v3 snmp-community SNMPV3COMMUNITY security-name NOSNMPV3
set snmp view SNMPVIEW oid .1 include
```

NOTE: The community used by the user which does not support SNMPv3, will continue to use SNMPv2.

For more information, see the following configuration:

```
snmpget -v 2c -c JTACCOMMUNITY 10.52.170.100 sysUpTime.0
```

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. For more information about this statement, see *community-name*.

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. For more information about this statement, see *context (SNMPv3)*.

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. For more information about this statement, see *tag*.

Example: Configure SNMPv3 Community

IN THIS SECTION

- Requirements | 568
- Overview | 569
- Configuration | 569
- Verification | 572

This example shows how to configure an SNMPv3 community.

Requirements

No special configuration beyond device initialization is required before configuring this example.

Overview

This example demonstrates how to create an SNMPv3 community. Define the SNMP community name, specify security name to perform the access control, and define tag name which identifies the address of managers that are allowed to use a community string. The target address defines a management application's address and parameters that are used in sending notifications.

When the device receives a packet with a recognized community string and a tag is associated with that packet, the Junos software 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.

Specify where you want the traps to be sent and define what SNMPv1 and SNMPv2c packets are allowed. Specify target address name that identifies the target address, define the target address, mask range of address, port number, tag list, and target parameter.

Configuration

IN THIS SECTION

- CLI Quick Configuration | 569
- Procedure | 570
- Results | 571

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit snmp v3]` hierarchy level, and then enter `commit` from configuration mode.

```
set snmp-community index1 community-name "public"
set snmp-community index1 security-name john
set snmp-community index1 tag router1
set target-address ta1 address 10.1.1.1
set target-address ta1 address-mask 255.255.255.0
set target-address ta1 port 162
set target-address ta1 tag-list router1
set target-address ta1 target-parameters tp1
```

Procedure

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

1. Configure the SNMP community name.

```
[edit snmp v3]
user@host# set snmp-community index1 community-name "public"
```

NOTE: The SNMP community name must be unique.

2. Configure the security name to perform access control.

```
[edit snmp v3]
user@host# set snmp-community index1 security-name john
```

3. Define the tag name. The tag name identifies the address of managers that are allowed to use a community string.

```
[edit snmp v3]
user@host# set snmp-community index1 tag router1
```

4. Configure SNMP target address.

```
[edit snmp v3]
user@host# set target-address ta1 address 10.1.1.1
```

5. Configure the mask range of the address for the community string access control.

```
[edit snmp v3]
user@host#set target-address ta1 address-mask 255.255.255.0
```

6. Configure SNMPv3 target port number.

```
[edit snmp v3]
user@host#set target-address ta1 port 162
```

7. Configure SNMPv3 tag list to select the target addresses.

```
[edit snmp v3]
user@host#set target-address ta1 tag-list router1
```

8. Configure SNMPv3 target parameter name in the target parameter table.

```
[edit snmp v3]
user@host#set target-address ta1 target-parameters tp1
```

Results

From configuration mode, confirm your configuration by entering the `show snmp v3` command. If the output does not display the intended configuration, repeat the configuration instructions in this example.

```
[edit]
user@host# show snmp v3
target-address ta1 {
    address 10.1.1.1;
    port 162;
    tag-list router1;
    address-mask 255.255.255.0;
    target-parameters tp1;
}
snmp-community index1 {
    community-name "$9$JOZi.QF/AtOz3"; ## SECRET-DATA
    security-name john;
    tag router1;
}
```

Verification

IN THIS SECTION

- Verifying SNMPv3 community | 572

Verifying SNMPv3 community

Purpose

Verify if SNMPv3 community is enabled.

Action

To verify SNMPv3 community configuration, enter `show snmp v3 community` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
Community           Security  Context  Tag          Storage      Status
index1              john              router1      nonvolatile  active
```

Meaning

The output displays the information about SNMPv3 community being enabled on the system.

MIB Views

IN THIS SECTION

- Configure MIB Views | 573
- Configure Ping Proxy MIB | 574

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.

Configure MIB Views

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, see *view (Configuring a MIB View)*.

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

```
[edit groups global snmp]
user@host# set view view-name oid object-identifier (include | exclude)
```

The following example creates a MIB view called ping-mib-view. The `oid` statement does not require a dot at the beginning of the object identifier. The `snmp view` statement includes the branch under the object identifier .1.3.6.1.2.1.80. This includes the entire DISMAN-PINGMIB subtree (as defined in RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*), which effectively permits access to any object under that branch.

```
[edit groups global snmp]
user@host# set view ping-mib-view oid 1.3.6.1.2.1.80 include
```

The following example adds a second branch in the same MIB view.

```
[edit groups global snmp]
user@host# set view ping-mib-view oid jnxPingMIB include
```

Assign a MIB view to a community that you want to control.

To associate MIB views with a community, see *view (SNMP Community)*.

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

SEE ALSO

oid

Configure Ping Proxy MIB

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;
}
```

SEE ALSO

view (Configuring a MIB View)

oid

SNMP MIBs Supported by Junos OS and Junos OS Evolved

IN THIS SECTION

- SNMP MIBs Support | 575
- MIB Objects for the QFX Series | 584
- Fabric Chassis MIB | 588
- Standard SNMP MIBs Supported by Junos OS | 594
- Enterprise-Specific MIBs for Junos OS Evolved | 609
- Enterprise-Specific SNMP MIBs Supported by Junos OS | 622
- Standard MIBs for Junos OS Evolved | 643

SNMP MIBs Support

IN THIS SECTION

- MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis | 575
- MIBs Supported on QFabric Systems | 581

The QFX Series standalone switches, QFX Series Virtual Chassis, and QFabric systems support standard MIBs and Juniper Networks enterprise-specific MIBs.

For information about enterprise-specific SNMP MIB objects, see the SNMP MIB Explorer. You can use SNMP MIB Explorer to view information about various MIBs, MIB objects, and SNMP notifications supported on Juniper Networks devices.

MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

The QFX Series standalone switches and QFX Series Virtual Chassis support both standard MIBs and Juniper Networks enterprise-specific MIBs. For more information, see:

- Table 48 on page 576 for standard MIBs.
- Table 49 on page 578 for Juniper Networks enterprise-specific MIBs.

Table 48: Standard MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

RFC	Additional Information
IEEE 802.1ab section 12.1, *Link Layer Discovery Protocol (LLDP) MIB*	Supported tables and objects: • lldpRemManAddrOID • lldpLocManAddrOID • lldpReinitDelay • lldpNotificationInterval • lldpStatsRxPortFramesDiscardedTotal • lldpStatsRxPortFramesError • lldpStatsRxPortTLVsDiscardedTotal • lldpStatsRxPortTLVsUnrecognizedTotal • lldpStatsRxPortAgeoutsTotal
IEEE 802.3ad, *Aggregation of Multiple Link Segments*	The following tables and objects are supported: • dot3adAggPortTable, dot3adAggPortListTable, dot3adAggTable, and dot3adAggPortStatsTable • dot3adAggPortDebugTable (only dot3adAggPortDebugRxState, dot3adAggPortDebugMuxState, dot3adAggPortDebugActorSyncTransitionCount, dot3adAggPortDebugPartnerSyncTransitionCount, dot3adAggPortDebugActorChangeCount, and dot3adAggPortDebugPartnerChangeCount) • dot3adTablesLastChanged
RFC 1286, *Definitions of Managed Objects for Bridges*	—

Table 48: Standard MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

RFC	Additional Information
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.
RFC 2933, *Internet Group Management Protocol (IGMP) MIB*	—
RFC 4318, *Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol*	Supports 802.1w and 802.1t extensions for RSTP. Not supported on OCX Series devices.
RFC 4363b, *Q-Bridge VLAN MIB*	**NOTE**: On QFX3500 and QFX3600 switches, the dot1dTpFdbTable table (RFC 4188, *Definitions of Managed Objects for Bridges*) is populated only with MAC addresses learned on the default VLAN. To see the MAC addresses of all VLANs, specify the dot1qTpFdbTable table (in this MIB) when you issue the `show snmp mib walk` command. Not supported on OCX Series devices.
Internet Assigned Numbers Authority, *IANAiftype Textual Convention MIB* (referenced by RFC 2233)	See http://www.iana.org/assignments/ianaiftype-mib .
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*	—
Internet draft draft-ietf-idmr-igmp-mib-13.txt, *Internet Group Management Protocol (IGMP) MIB*	—

Table 48: Standard MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

RFC	Additional Information
ESO Consortium MIB	**NOTE**: The ESO Consortium MIB has been replaced by RFC 3826. See http://www.snmp.com/eso/.

Table 49: Juniper Networks Enterprise-Specific MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis

MIB	Description
Alarm MIB (mib-jnx-chassis-alarm)	Provides support for alarms from the switch.
Analyzer MIB (mib-jnx-analyzer)	Contains analyzer and remote analyzer data related to *port mirroring*. Not supported on OCX Series devices.
Chassis MIB (mib-jnx-chassis)	Provides support for environmental monitoring (power supply state, board voltages, fans, temperatures, and airflow) and inventory support for the chassis, Flexible PIC Concentrators (FPCs), and PICs. **NOTE**: The jnxLEDTable table has been deprecated.
Chassis Definitions for Router Model MIB (mib-jnx-chas-defines)	Contains the object identifiers (OIDs) that are used by the Chassis MIB to identify routing and switching platforms and chassis components. The Chassis MIB provides information that changes often, whereas the Chassis Definitions for Router Model MIB provides information that changes less often.
Class-of-Service MIB (mib-jnx-cos)	Provides support for monitoring interface output queue statistics per interface and per forwarding class.
Configuration Management MIB (mib-jnx-cfgmgmt)	Provides notification for configuration changes and rescue configuration changes in the form of 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.

Table 49: Juniper Networks Enterprise-Specific MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

MIB	Description
Ethernet MAC MIB (mib-jnx-mac)	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. Not supported on OCX Series devices.
Event MIB (mib-jnx-event)	Defines a generic trap that can be generated using an operations 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. In Junos OS release 13.2X51-D10 or later, if you configured an event policy to raise a trap when a new SNMP trap target is added, the SNMPD_TRAP_TARGET_ADD_NOTICE trap is generated with information about the new target.
Firewall MIB (mib-jnx-firewall)	Provides support for monitoring *firewall filter* counters.
Host Resources MIB (mib-jnx-hostresources)	Extends the hrStorageTable object, providing a measure of the usage of each file system on the switch as a 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.
Interface MIB (Extensions) (mib-jnx-if-extensions)	Extends the standard ifTable (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information in the ifJnxTable and ifChassisTable tables.
L2ALD MIB (mib-jnx-l2ald)	Provides information about Layer 2 Address Learning and related traps, such as the routing instance MAC limit trap and interface MAC limit trap. This MIB also provides VLAN information in the jnxL2aldVlanTable table for Enhanced Layer 2 Software (ELS) EX Series and QFX Series switches. **NOTE**: Non-ELS EX Series switches use the VLAN MIB (jnxExVlanTable) for VLAN information instead of this MIB.

Table 49: Juniper Networks Enterprise-Specific MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

MIB	Description
MPLS MIB (mib-jnx-mpls)	Provides MPLS information and defines MPLS notifications. **NOTE**: This MIB is not supported on the QFX5100 switch.
MPLS LDP MIB (mib-jnx-mpls-ldp)	Contains object definitions as described in RFC 3815, *Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP)*. **NOTE**: This MIB is not supported on the QFX5100 switch.
Ping MIB (mib-jnx-ping)	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.
RMON Events and Alarms MIB (mib-jnx-rmon)	Supports Junos OS extensions to the standard Remote Monitoring (RMON) Events and Alarms MIB (RFC 2819). The extension augments the alarmTable object with additional information about each alarm. Two additional traps are also defined to indicate when problems are encountered with an alarm.
Structure of Management Information MIB (mib-jnx-smi)	Explains how the Juniper Networks enterprise-specific MIBs are structured.
System Log MIB (mib-jnx-syslog)	Enables notification of an SNMP trap-based application when an important system log message occurs.
Utility MIB (mib-jnx-util)	Provides you with SNMP MIB container objects of the following types: 32-bit counters, 64-bit counters, signed integers, unsigned integers, and octet strings. You can use these objects to store data that can be retrieved using other SNMP operations.

Table 49: Juniper Networks Enterprise-Specific MIBs Supported on QFX Series Standalone Switches and QFX Series Virtual Chassis *(Continued)*

MIB	Description
VLAN MIB (mib-jnx-vlan)	Contains information about prestandard IEEE 802.10 VLANs and their association with LAN emulation clients. **NOTE**: For ELS EX Series switches and QFX Series switches, VLAN information is available in the L2ALD MIB in the jnxL2aldVlanTable table instead of in the VLAN MIB For non-ELS EX Series switches, VLAN information is provided in the VLAN MIB in the jnxExVlanTable table. Not supported on OCX Series devices.

MIBs Supported on QFabric Systems

The QFabric systems support both standard MIBs and Juniper Networks enterprise-specific MIBs. For more information, see:

- Table 50 on page 581 for standard MIBs.
- Table 51 on page 582 for Juniper Networks enterprise-specific MIBs.

Table 50: Standard MIBs Supported on QFabric Systems

RFC	Additional Information
RFC 1286, *Definitions of Managed Objects for Bridges*	—
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.
RFC 2933, *Internet Group Management Protocol (IGMP) MIB*	—

Table 50: Standard MIBs Supported on QFabric Systems *(Continued)*

RFC	Additional Information
RFC 4363b, *Q-Bridge VLAN MIB*	The QFabric system supports the following tables only: • dot1qTpFdbTable • dot1qVlanStaticTable • dot1qPortVlanTable • dot1qFdbTable Not supported on OCX Series devices.

NOTE: QFabric-specific MIBs are not supported on OCX Series devices.

Table 51: Juniper Networks Enterprise-Specific MIBs Supported on QFabric Systems

MIB	Description
Analyzer MIB (mib-jnx-analyzer)	Contains analyzer and remote analyzer data related to port mirroring. The QFabric system supports: • Analyzer table—jnxAnalyzerName, jnxMirroringRatio, jnxLossPriority. • Analyzer input table—jnxAnalyzerInputValue, jnxAnalyzerInputOption, jnxAnalyzerInputType. • Analyzer output table—jnx AnalyzerOutputValue, jnxAnalyzerOutputType.
Chassis MIB (mib-jnx-chassis)	**NOTE**: The Chassis MIB has been deprecated for the QFabric system. We recommend that you use the Fabric Chassis MIB (mib-jnx-fabric-chassis) for information about the QFabric system.

Table 51: Juniper Networks Enterprise-Specific MIBs Supported on QFabric Systems *(Continued)*

MIB	Description
Class-of-Service MIB (mib-jnx-cos)	Provides support for monitoring interface output queue statistics per interface and per forwarding class. The QFabric system supports the following tables and objects: • Jnxcosifstatflagtable—jnxCosIfstatFlags and jnxCosIfIndex. • Jnxcosqstattable—jnxCosQstatTxedPkts, jnxCosQstatTxedPktRate, jnxCosQstatTxedBytes, and jnxCosQstatTxedByteRate. • Jnxcosfcidtable—jnxCosFcIdToFcName. • Jnxcosfctable—jnxCosFcQueueNr. The QFabric system does not support any traps for this MIB.
Configuration Management MIB (mib-jnx-cfgmgmt)	Provides notification for configuration changes and rescue configuration changes in the form of 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. **NOTE**: On the QFabric system, these conditions apply: • All scalar variables under the jnxCmCfgChg table are supported. • Supported scalar OIDs are jnxCmCfgChgLatestIndex, jnxCmCfgChgLatestTime, jnxCmCfgChgLatestDate, jnxCmCfgChgLatestSource, jnxCmCfgChgLatestUser, and jnxCmCfgChgMaxEventEntries. • Scalar variables under the jnxCmRescueChg table are not supported.
Fabric Chassis MIB (mib-jnx-fabric-chassis)	Provides hardware information about the QFabric system and its component devices. This MIB is based on the Juniper Networks enterprise-specific Chassis MIB but adds another level of indexing that provides information for QFabric system component devices.

Table 51: Juniper Networks Enterprise-Specific MIBs Supported on QFabric Systems *(Continued)*

MIB	Description
Interface MIB (Extensions) (mib-jnx-if-extensions)	Extends the standard ifTable (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information in the ifJnxTable and ifChassisTable tables. **NOTE**: On the QFabric system, scalar variables are not supported.
Power Supply Unit MIB (mib-jnx-power-supply-unit)	Provides support for environmental monitoring of the power supply unit for the Interconnect device of the QFabric system. **NOTE**: On the QFabric system, scalar variables for the jnxPsuObjects 1 object ID in the jnxPsuScalars table are not supported.
QFabric MIB (jnx-qf-smi)	Explains how the Juniper Networks enterprise-specific QFabric MIBs are structured. Defines the MIB objects that are reported by the QFabric system and the contents of the traps that can be issued by the QFabric system.
Utility MIB (mib-jnx-util)	Provides you with SNMP MIB container objects of the following types: 32-bit counters, 64-bit counters, signed integers, unsigned integers, and octet strings. You can use these objects to store data that can be retrieved using other SNMP operations.

SEE ALSO

SNMP MIB Explorer

Understanding the Implementation of SNMP on the QFabric System

MIB Objects for the QFX Series

IN THIS SECTION

- QFX Series Standalone Switches | **585**
- QFabric Systems | **585**

- QFabric System QFX3100 Director Device | **586**
- QFabric System QFX3008-I Interconnect Device | **586**
- QFabric System QFX3600-I Interconnect Device | **587**
- QFabric System Node Devices | **587**

This topic lists the Juniper Networks enterprise-specific SNMP Chassis MIB definition objects for the QFX Series:

QFX Series Standalone Switches

```
jnxProductLineQFXSwitch       OBJECT IDENTIFIER ::= { jnxProductLine      82 }
  jnxProductNameQFXSwitch       OBJECT IDENTIFIER ::= { jnxProductName      82 }
  jnxProductModelQFXSwitch      OBJECT IDENTIFIER ::= { jnxProductModel     82 }
  jnxProductVariationQFXSwitch  OBJECT IDENTIFIER ::= { jnxProductVariation 82 }
    jnxProductQFX3500s          OBJECT IDENTIFIER ::= { jnxProductVariationQFXSwitch 1 }
    jnxProductQFX360016QS       OBJECT IDENTIFIER ::= { jnxProductVariationQFXSwitch 2 }
    jnxProductQFX350048T4QS     OBJECT IDENTIFIER ::= { jnxProductVariationQFXSwitch 3 }
    jnxProductQFX510024Q        OBJECT IDENTIFIER ::= { jnxProductVariationQFXSwitch 4 }
    jnxProductQFX510048S6Q      OBJECT IDENTIFIER ::= { jnxProductVariationQFXSwitch 5 }

  jnxChassisQFXSwitch           OBJECT IDENTIFIER ::= { jnxChassis          82 }

  jnxSlotQFXSwitch              OBJECT IDENTIFIER ::= { jnxSlot             82 }
    jnxQFXSwitchSlotFPC         OBJECT IDENTIFIER ::= { jnxSlotQFXSwitch   1 }
    jnxQFXSwitchSlotHM          OBJECT IDENTIFIER ::= { jnxSlotQFXSwitch   2 }
    jnxQFXSwitchSlotPower       OBJECT IDENTIFIER ::= { jnxSlotQFXSwitch   3 }
    jnxQFXSwitchSlotFan         OBJECT IDENTIFIER ::= { jnxSlotQFXSwitch   4 }
    jnxQFXSwitchSlotFPB         OBJECT IDENTIFIER ::= { jnxSlotQFXSwitch   5 }

  jnxMediaCardSpaceQFXSwitch    OBJECT IDENTIFIER ::= { jnxMediaCardSpace   82 }
    jnxQFXSwitchMediaCardSpacePIC OBJECT IDENTIFIER ::= { jnxMediaCardSpaceQFXSwitch 1 }
```

QFabric Systems

```
jnxProductLineQFX3000        OBJECT IDENTIFIER ::= { jnxProductLine 84 }
  jnxProductNameQFX3000         OBJECT IDENTIFIER ::= { jnxProductName 84 }
```

```
  jnxProductModelQFX3000        OBJECT IDENTIFIER ::= { jnxProductModel 84 }
  jnxProductVariationQFX3000    OBJECT IDENTIFIER ::= { jnxProductVariation 84 }
   jnxProductQFX3000-G          OBJECT IDENTIFIER ::= { jnxProductVariationQFX3000 1 }
   jnxProductQFX3000-M          OBJECT IDENTIFIER ::= { jnxProductVariationQFX3000 2 }
  jnxChassisQFX3000             OBJECT IDENTIFIER ::= { jnxChassis          84 }
```

QFabric System QFX3100 Director Device

```
jnxProductLineQFX3100 OBJECT IDENTIFIER ::= { jnxProductLine      100 }
  jnxProductNameQFX3100 OBJECT IDENTIFIER ::= { jnxProductName      100 }
  jnxProductModelQFX3100 OBJECT IDENTIFIER ::= { jnxProductModel     100 }
  jnxProductVariationQFX3100  OBJECT IDENTIFIER ::= { jnxProductVariation 100 }
  jnxChassisQFX3100     OBJECT IDENTIFIER ::= { jnxChassis          100 }

  jnxSlotQFX3100            OBJECT IDENTIFIER ::= { jnxSlot             100 }
    jnxQFX3100SlotCPU       OBJECT IDENTIFIER ::= { jnxSlotQFX3100   1 }
    jnxQFX3100SlotMemory    OBJECT IDENTIFIER ::= { jnxSlotQFX3100   2 }
    jnxQFX3100SlotPower     OBJECT IDENTIFIER ::= { jnxSlotQFX3100   3 }
    jnxQFX3100SlotFan       OBJECT IDENTIFIER ::= { jnxSlotQFX3100   4 }
    jnxQFX3100SlotHardDisk  OBJECT IDENTIFIER ::= { jnxSlotQFX3100   5 }
    jnxQFX3100SlotNIC       OBJECT IDENTIFIER ::= { jnxSlotQFX3100   6 }
```

QFabric System QFX3008-I Interconnect Device

```
jnxProductLineQFXInterconnect OBJECT IDENTIFIER ::= { jnxProductLine      60 }
  jnxProductNameQFXInterconnect OBJECT IDENTIFIER ::= { jnxProductName      60 }
  jnxProductModelQFXInterconnect OBJECT IDENTIFIER ::= { jnxProductModel     60 }
  jnxProductVariationQFXInterconnect  OBJECT IDENTIFIER ::= { jnxProductVariation 60 }
    jnxProductQFX3008           OBJECT IDENTIFIER ::= { jnxProductVariationQFXInterconnect
1 }    jnxProductQFXC083008        OBJECT IDENTIFIER ::= { jnxProductVariationQFXInterconnect 2 }
    jnxProductQFX3008I          OBJECT IDENTIFIER ::= { jnxProductVariationQFXInterconnect 3 }

  jnxChassisQFXInterconnect     OBJECT IDENTIFIER ::= { jnxChassis          60 }

  jnxSlotQFXInterconnect        OBJECT IDENTIFIER ::= { jnxSlot             60 }
    jnxQFXInterconnectSlotFPC   OBJECT IDENTIFIER ::= { jnxSlotQFXInterconnect   1 }
    jnxQFXInterconnectSlotHM    OBJECT IDENTIFIER ::= { jnxSlotQFXInterconnect   2 }
    jnxQFXInterconnectSlotPower OBJECT IDENTIFIER ::= { jnxSlotQFXInterconnect   3 }
    jnxQFXInterconnectSlotFan   OBJECT IDENTIFIER ::= { jnxSlotQFXInterconnect   4 }
```

```
    jnxQFXInterconnectSlotCBD   OBJECT IDENTIFIER ::= { jnxSlotQFXInterconnect   5 }
    jnxQFXInterconnectSlotFPB   OBJECT IDENTIFIER ::= { jnxSlotQFXInterconnect   6 }

  jnxMediaCardSpaceQFXInterconnect    OBJECT IDENTIFIER ::= { jnxMediaCardSpace   60 }
    jnxQFXInterconnectMediaCardSpacePIC OBJECT IDENTIFIER ::= { jnxMediaCardSpaceQFXInterconnect
1 }

  jnxMidplaneQFXInterconnect    OBJECT IDENTIFIER ::= { jnxBackplane        60 }
```

QFabric System QFX3600-I Interconnect Device

```
jnxProductLineQFXMInterconnect OBJECT IDENTIFIER ::= { jnxProductLine      91 }
  jnxProductNameQFXMInterconnect OBJECT IDENTIFIER ::= { jnxProductName      91 }
  jnxProductModelQFXMInterconnect OBJECT IDENTIFIER ::= { jnxProductModel     91 }
  jnxProductVariationQFXMInterconnect  OBJECT IDENTIFIER ::= { jnxProductVariation 91 }
    jnxProductQFX3600I          OBJECT IDENTIFIER ::= { jnxProductVariationQFXMInterconnect 1 }

  jnxChassisQFXMInterconnect     OBJECT IDENTIFIER ::= { jnxChassis          91 }

  jnxSlotQFXMInterconnect        OBJECT IDENTIFIER ::= { jnxSlot             91 }
    jnxQFXMInterconnectSlotFPC   OBJECT IDENTIFIER ::= { jnxSlotQFXMInterconnect   1 }
    jnxQFXMInterconnectSlotHM    OBJECT IDENTIFIER ::= { jnxSlotQFXMInterconnect   2 }
    jnxQFXMInterconnectSlotPower OBJECT IDENTIFIER ::= { jnxSlotQFXMInterconnect   3 }
    jnxQFXMInterconnectSlotFan   OBJECT IDENTIFIER ::= { jnxSlotQFXMInterconnect   4 }
    jnxQFXMInterconnectSlotFPB   OBJECT IDENTIFIER ::= { jnxSlotQFXMInterconnect   5 }

  jnxMediaCardSpaceQFXMInterconnect    OBJECT IDENTIFIER ::= { jnxMediaCardSpace   91 }
    jnxQFXMInterconnectMediaCardSpacePIC OBJECT IDENTIFIER ::=
{ jnxMediaCardSpaceQFXMInterconnect 1 }
```

QFabric System Node Devices

```
jnxProductLineQFXNode       OBJECT IDENTIFIER ::= { jnxProductLine      61 }
  jnxProductNameQFXNode       OBJECT IDENTIFIER ::= { jnxProductName      61 }
  jnxProductModelQFXNode      OBJECT IDENTIFIER ::= { jnxProductModel     61 }
  jnxProductVariationQFXNode  OBJECT IDENTIFIER ::= { jnxProductVariation 61 }
    jnxProductQFX3500         OBJECT IDENTIFIER ::= { jnxProductVariationQFXNode 1 }
    jnxProductQFX360016Q      OBJECT IDENTIFIER ::= { jnxProductVariationQFXNode 3 }
```

```
jnxChassisQFXNode           OBJECT IDENTIFIER ::= { jnxChassis          61 }

jnxSlotQFXNode              OBJECT IDENTIFIER ::= { jnxSlot             61 }
  jnxQFXNodeSlotFPC         OBJECT IDENTIFIER ::= { jnxSlotQFXNode   1 }
  jnxQFXNodeSlotHM          OBJECT IDENTIFIER ::= { jnxSlotQFXNode   2 }
  jnxQFXNodeSlotPower       OBJECT IDENTIFIER ::= { jnxSlotQFXNode   3 }
  jnxQFXNodeSlotFan         OBJECT IDENTIFIER ::= { jnxSlotQFXNode   4 }
  jnxQFXNodeSlotFPB         OBJECT IDENTIFIER ::= { jnxSlotQFXNode   5 }

jnxMediaCardSpaceQFXNode    OBJECT IDENTIFIER ::= { jnxMediaCardSpace   61 }
  jnxQFXNodeMediaCardSpacePIC OBJECT IDENTIFIER ::= { jnxMediaCardSpaceQFXNode 1 }
```

SEE ALSO

> *Understanding the Implementation of SNMP on the QFabric System*

Fabric Chassis MIB

The Juniper Networks enterprise-specific SNMP Fabric Chassis MIB (mib-jnx-fabric-chassis) provides hardware information about the QFabric system and its component devices in a single MIB. The Fabric Chassis MIB is based on the Juniper Networks enterprise-specific Chassis MIB that provides information for individual devices. Unlike the Chassis MIB, the Fabric Chassis MIB represents the QFabric system component devices as part of the QFabric system. Only the information from the Fabric Chassis MIB (and not from individual Chassis MIBs) is available to SNMP management clients of the QFabric system.

The Fabric Chassis MIB uses the basic information structure of the Chassis MIB, but adds another level of indexing that provides detailed information about QFabric system devices. Each physical device in a QFabric system (such as a Node device or an Interconnect device) is represented with its hardware components, including the power supply, fans, and front and rear cards.

As in other SNMP systems, the SNMP manager resides on the network management system (NMS) of the network to which the QFabric system belongs. The SNMP agent (snmpd) resides in the QFabric system Director software and is responsible for receiving and distributing all traps as well as responding to all queries from the SNMP manager.

In addition, there is an SNMP subagent running in the Routing Engine of each Node group and Interconnect device. The SNMP subagent manages the information about the component device, and that information is communicated to the SNMP agent in the Director software as needed. Traps that are generated by a Node device are sent to the SNMP agent in the Director software, which in turn processes and sends them to the target IP addresses that are defined in the SNMP configuration.

Table 52 on page 589 describes the tables and objects in the Fabric Chassis MIB.

Table 52: Fabric Chassis MIB Tables and Objects

Table or Object Name	Root OID	Description
Tables with Counterparts in the Chassis MIB		
jnxFabricContainersTable	1.3.6.1.4.1.2636.3.42.2.2.2	Provides information about different types of containers in QFabric system devices. • Containers for Interconnect devices include fan trays, power supply units, control boards, and so on. • Containers for Node devices include fan trays, power supply units, Flexible PIC Concentrator (FPC), PICs, and so on. • Containers for the Director devices include CPU, memory, fan trays, power supply units, and hard disks. The containers have a non-hierarchical or flat structure, and components in them are organized as siblings to each other.
jnxFabricContentsTable	1.3.6.1.4.1.2636.3.42.2.2.3	Contains contents that are present across all devices represented in the jnxFabricDeviceTable object. This table includes all field replaceable units (FRUs) and non-FRUs for QFabric system devices. • Contents in the Interconnect devices include fan trays and control boards. • Contents in the Node devices include fan trays and power supply units. • Contents in the Director devices include CPUs, memory, fan trays, power supply units, and hard disks, but do not include network interface cards (NICs).

Table 52: Fabric Chassis MIB Tables and Objects *(Continued)*

Table or Object Name	Root OID	Description
jnxFabricFilledTable	1.3.6.1.4.1.2636.3.42.2.2.4	Shows the status of containers in QFabric devices. The jnxFabricFilledState object represents the state of the component: (1) unknown, (2) empty, or (3) filled. **NOTE**: The jnxFabricFilledTable object does not contain information about the Director group.
jnxFabricOperatingTable	1.3.6.1.4.1.2636.3.42.2.2.5	Represents different operating parameters for the contents that are populated in the jnxFabricContentsTable object. • Contents in each Node device and Interconnect device include fan trays, power supply units, FPC, PIC, and Routing Engine. • Contents in the Director device include CPUs, memory, fan trays, power supply units, and hard disks, but do not include network interface cards (NICs). The jnxFabricOperatingState object provides the state of the device: (1) unknown, (2) running, (3) ready, (4) reset, (5) runningAtFullSpeed (for fans only), (6) down, (6) off (for power supply units), or (7) standby.
jnxFabricRedundancyTable	1.3.6.1.4.1.2636.3.42.2.2.6	Represents the redundancy information that is available at different subsystem levels across the QFabric system. Information about the Routing Engines in Node devices is included, but there are no corresponding entries for Interconnect devices in this table. The jnxFabricRedundancyState object indicates the state of the subsystem: (1) unknown, (2) primary, (3) backup, or (4) disabled. **NOTE**: Information about redundant Director devices, virtual machines (VMs) within Director groups, and Virtual Chassis devices is not available at this time.

Table 52: Fabric Chassis MIB Tables and Objects *(Continued)*

Table or Object Name	Root OID	Description
jnxFabricFruTable	1.3.6.1.4.1.2636.3.42.2.2.7	Contains all FRUs for the QFabric system in the jnxFabricDeviceTable table. The FRUs are listed regardless of whether or not they are installed or online. The jnxFabricFruState object represents the state of the FRU, including online, offline, or empty, and so on. This table also contains information about each FRU, such as name, type, temperature, time last powered on, and time last powered off. **NOTE**: The jnxFabricFruTable table does not include network interface cards (NICs) on Director devices.
Table Specific to the Fabric Chassis MIB		
jnxFabricDeviceTable	1.3.6.1.4.1.2636.3.42.2.2.1	Contains information about all devices in the QFabric system. This table organizes scalar variables represented in the Chassis MIB into a table format for the QFabric system component devices. Columns in this table include device information such as model, device alias, and serial number. The jnxFabricDeviceIndex identifies each QFabric system device (Node device, Interconnect device, and Director device). **NOTE**: At this time, information about the Virtual Chassis is not available. **NOTE**: The following objects are not supported: • jnxFabricDeviceEntryRevision • jnxFabricDeviceEntryFirmwareRevision • jnxFabricDeviceEntryKernelMemoryUsedPercent
Scalar Variables		

Table 52: Fabric Chassis MIB Tables and Objects *(Continued)*

Table or Object Name	Root OID	Description
The following scalar variables are supported: • jnxFabricClass • jnxFabricDescr • jnxFabricSerialNo • jnxFabricRevision • jnxFabricLastInstalled • jnxFabricContentsLastChange • jnxFabricFilledLastChange	1.3.6.1.4.1.2636.3.42.2.1	Describe the QFabric system as a whole. **NOTE**: The jnxFabricFirmwareRevision scalar variable is not supported at this time.

Table 53 on page 593 describes the SNMPv2 traps that are defined in the Fabric Chassis MIB.

NOTE: Only SNMPv2 traps are supported on the QFabric system.

Table 53: Fabric Chassis MIB SNMPv2 Traps

Trap Group and Name	Root OID	Description
jnxFabricChassisTraps group—Includes the following traps: • jnxFabricPowerSupplyFailure • jnxFabricFanFailure • jnxFabricOverTemperature • jnxFabricRedundancySwitchover • jnxFabricFruRemoval • jnxFabricFruInsertion • jnxFabricFruPowerOff • jnxFabricFruPowerOn • jnxFabricFruFailed • jnxFabricFruOffline • jnxFabricFruOnline • jnxFabricFruCheck • jnxFabricFEBSwitchover • jnxFabricHardDiskFailed • jnxFabricHardDiskMissing • jnxFabricBootFromBackup • jnxFabricHighPower	1.3.6.1.4.1.2636.4.19	Indicates an alarm condition. **NOTE**: Hardware events on the Director group are detected by scanning. As a result, a trap may not be generated until up to 30 seconds after the event has occurred. **NOTE**: The software does not distinguish between the fan removal and fan failure events on the Director group. In each case, both the jnxFabricFanFailure and jnxFabricFruFailed traps are generated.

Table 53: Fabric Chassis MIB SNMPv2 Traps *(Continued)*

Trap Group and Name	Root OID	Description
jnxFabricChassisOKTraps group—Includes the following traps: • jnxFabricPowerSupplyOK • jnxFabricFanOK • jnxFabricTemperatureOK • jnxFabricFruOK • jnxFabricHighPowerCleared	1.3.6.1.4.1.2636.4.20	Indicates an alarm cleared condition.

SEE ALSO

Understanding the Implementation of SNMP on the QFabric System

Standard SNMP MIBs Supported by Junos OS

Junos OS supports the Standard MIBs listed in Table 54 on page 594.

Table 54: Standard MIBs supported by Junos OS

Standard MIB	Supported and unsupported tables and objects	Platforms
IEEE 802.1ab section 12.1, *Link Layer Discovery Protocol (LLDP) MIB*	EX Series implementation of LLDP MIB supports both IPv4 and IPv6 configuration.	EX Series and MX Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
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	EX Series, M Series, MX Series, PTX Series, SRX Series, T Series, and vSRX

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
IEEE, 802.1ag, *Connectivity Fault Management*	Supported tables and objects: • dot1agCfmMdTableNextIndex • dot1agCfmMdTable (except dot1agCfmMdMhfldPermission) • 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)	EX Series, MX Series, PTX Series, and QFX Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
IEEE, 802.1ap, *Management Information Base (MIB) definitions for VLAN Bridges*	Supported tables and objects: • `ieee8021CfmStackTable` • `ieee8021CfmVlanTable` • `ieee8021CfmDefaultMdTable` (except `ieee8021CfmDefaultMdIdPermission`) • `ieee8021CfmMaCompTable` (except `ieee8021CfmMaCompIdPermission`)	MX Series
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.	No exceptions	ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, and T Series
RFC 2922, *The Physical Topology (PTOPO) MIB*	Supported objects: • `ptopoConnDiscAlgorithm` • `ptopoConnAgentNetAddrType` • `ptopoConnAgentNetAddr` • `ptopoConnMultiMacSASeen` • `ptopoConnMultiNetSASeen` • `ptopoConnIsStatic` • `ptopoConnLastVerifyTime` • `ptopoConnRowStatus`	EX Series and SRX Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
RFC 3591 *Managed Objects for the Optical Interface Type*	Supported tables and objects: • `optIfOTMnTable` (except `optIfOTMnOpticalReach`, `optIfOTMnInterfaceType`, and `optIfOTMnOrder`) • `optIfOChConfigTable` (except `optIfOChDirectionality` and `optIfOChCurrentStatus`) • `optIfOTUkConfigTable` (except `optIfOTUkTraceIdentifierAccepted`, `optIfOTUkTIMDetMode`, `optIfOTUkTIMActEnabled`, `optIfOTUkTraceIdentifierTransmitted`, `optIfOTUkDEGThr`, `optIfOTUkDEGM`, `optIfOTUkSinkAdaptActive`, and `optIfOTUkSourceAdaptActive`) • `optIfODUkConfigTable` (except `optIfODUkPositionSeqCurrentSize` and `optIfODUkTtpPresent`)	M Series, MX Series, PTX Series, and T Series
RFC 3621, *Power Ethernet MIB*	No exceptions	EX Series
RFC 3637, *Definitions of Managed Objects for the Ethernet WAN Interface Sublayer*	Unsupported tables and objects: • `etherWisDeviceTable`, • `etherWisSectionCurrentTable` • `etherWisFarEndPathCurrentTable`	M Series, MX Series, PTX Series, and T Series
RFC 3877, *Alarm Management Information Base*	• 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.	MX Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
RFC 3896, *Definitions of Managed Objects for the DS3/E3 Interface Type*	Unsupported tables and objects: • `dsx3FarEndConfigTable` • `dsx3FarEndCurrentTable` • `dsx3FarEndIntervalTable` • `dsx3FarEndTotalTable` • `dsx3FracTable`	M Series and T Series
RFC 4318, *Definitions of Managed Objects for Bridges with Rapid Spanning Tree Protocol*	Supports 802.1w and 802.1t extensions for RSTP.	EX Series, M Series, MX Series, and T Series
RFC 4363b, *Q-Bridge VLAN MIB*	No exceptions	MX Series and EX Series
RFC 4668, *RADIUS Accounting Client Management Information Base (MIB) for IPv6* (read-only access)	No exceptions	MX Series
RFC 4670, *RADIUS Accounting Client Management Information Base (MIB)* (read-only access)	No exceptions	MX Series
RFC 4801, *Definitions of Textual Conventions for Generalized Multiprotocol Label Switching (GMPLS) Management Information Base (MIB)* (read-only access)	No exceptions	M Series, MX Series, and T Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
RFC 4802, *Generalized Multiprotocol Label Switching (GMPLS) Traffic Engineering (TE) Management Information Base (MIB)* (read-only access)	Unsupported tables and objects: • `gmplsTunnelReversePerfTable` • `gmplsTeScalars` • `gmplsTunnelTable` • `gmplsTunnelARHopTable` • `gmplsTunnelCHopTable` • `gmplsTunnelErrorTable`	M Series, MX Series, and T Series
RFC 4803, *Generalized Multiprotocol Label Switching (GMPLS) Label Switching Router (LSR) Management Information Base (MIB)* (read-only access) **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.	Unsupported tables and objects: • `gmplsLabelTable` • `gmplsOutsegmentTable`	M Series, MX Series, and T Series
RFC 5132, *IP Multicast MIB* **NOTE**: This RFC obsoletes RFC2932.	Unsupported table: • `ipMcastZoneTable`	All platforms

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
RFC 5643, *Management Information Base for OSPFv3* (read-only access)	Unsupported tables and objects: • ospfv3HostTable • ospfv3CfgNbrTable • ospfv3ExitOverflowInterval • ospfv3ReferenceBandwidth • ospfv3RestartSupport • ospfv3RestartInterval • ospfv3RestartStrictLsaChecking • ospfv3RestartStatus • ospfv3RestartAge • ospfv3RestartExitReason • ospfv3NotificationEnable • ospfv3StubRouterSupport • ospfv3StubRouterAdvertisement • ospfv3DiscontinuityTime • ospfv3RestartTime • ospfv3AreaNssaTranslatorRole • ospfv3AreaNssaTranslatorState • ospfv3AreaNssaTranslatorStabInterval • ospfv3AreaNssaTranslatorEvents • ospfv3AreaTEEnabled • ospfv3IfMetricValue	M Series, MX Series, PTX Series, SRX Series, and T Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
	• ospfv3IfDemandNbrProbe	

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
RFC 7420, *Path Computation Element Communication*	The PCEP MIB module is limited to "read-only" access except for pcePcepNotificationsMaxRate, which is used to throttle the rate at which the implementation generates notifications. In the mentioned tables only PCEP peer and PCEP session table will be supported in this release. For `pcePcepPeerTable`, the following members are not supported: • `pcePcepPeerDiscontinuityTime TimeStamp,` • `pcePcepPeerLWMRspTime Unsigned32,` • `pcePcepPeerHWMRspTime Unsigned32,` • `pcePcepPeerNumPCReqSent Counter32,` • `pcePcepPeerNumPCReqRcvd Counter32,` • `pcePcepPeerNumPCRepSent Counter32,` • `pcePcepPeerNumPCRepRcvd Counter32,` • `pcePcepPeerAvgRspTime Unsigned32,` • `pcePcepPeerNumReqSent Counter32,` • `pcePcepPeerNumReqSentEroRcvd Counter32,` • `pcePcepPeerNumReqSentErrorRcvd Counter32,` • `pcePcepPeerNumReqSentTimeout Counter32,` • `pcePcepPeerNumReqSentPendRep Counter32,` • `pcePcepPeerNumReqSentCancelSent Counter32,` • `pcePcepPeerNumReqSentClosed Counter32,` • `pcePcepPeerNumReqRcvd Counter32,` • `pcePcepPeerNumPCNtfSent Counter32,`	MX Series and PTX Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
	• pcePcepPeerNumPCNtfRcvd Counter32, • pcePcepPeerNumSvecSent Counter32, • pcePcepPeerNumSvecReqSent Counter32, • pcePcepPeerNumSvecRcvd Counter32, • pcePcepPeerNumSvecReqRcvd Counter32, • pcePcepPeerNumReqRcvdPendRep Counter32, • pcePcepPeerNumReqRcvdEroSent Counter32, • pcePcepPeerNumReqRcvdNoPathSent Counter32, • pcePcepPeerNumReqRcvdCancelSent Counter32, • pcePcepPeerNumReqRcvdErrorSent Counter32, • pcePcepPeerNumReqRcvdCancelRcvd Counter32, • pcePcepPeerNumReqRcvdClosed Counter32, • pcePcepPeerNumRepRcvdUnknown Counter32, • pcePcepPeerNumReqRcvdUnknown Counter32, • pcePcepPeerNumReqSentNoPathRcvd Counter32, • pcePcepPeerNumReqSentCancelRcvd Counter32	

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
	For pcePcepSessTable, the following members are not supported: • pcePcepSessNumPCReqSent Counter32, • pcePcepSessNumPCReqRcvd Counter32, • pcePcepSessKAHoldTimeRem Unsigned32, • pcePcepSessOverloaded TruthValue, • pcePcepSessOverloadTime Unsigned32, • pcePcepSessPeerOverloaded TruthValue, • pcePcepSessPeerOverloadTime Unsigned32, • pcePcepSessNumPCNtfSent Counter32, • pcePcepSessNumPCNtfRcvd Counter32, • pcePcepSessNumReqSent Counter32, • pcePcepSessNumReqSentPendRep Counter32, • pcePcepSessNumReqSentEroRcvd Counter32, • pcePcepSessNumReqSentNoPathRcvd Counter32, • pcePcepSessNumReqSentCancelRcvd Counter32, • pcePcepSessNumReqSentErrorRcvd Counter32, • pcePcepSessNumReqSentTimeout Counter32, • pcePcepSessNumReqSentCancelSent Counter32, • pcePcepSessAvgRspTime Unsigned32, • pcePcepSessLWMRspTime Unsigned32, • pcePcepSessHWMRspTime Unsigned32,	

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
	• pcePcepSessNumSvecSent Counter32, • pcePcepSessNumSvecReqSent Counter32, • pcePcepSessNumReqRcvd Counter32, • pcePcepSessNumSvecRcvd Counter32, • pcePcepSessNumSvecReqRcvd Counter32, • pcePcepSessNumReqRcvdPendRep Counter32, • pcePcepSessNumReqRcvdEroSent Counter32, • pcePcepSessNumReqRcvdNoPathSent Counter32, • pcePcepSessNumReqRcvdCancelSent Counter32, • pcePcepSessNumReqRcvdErrorSent Counter32, • pcePcepSessNumReqRcvdCancelRcvd Counter32, • pcePcepSessNumRepRcvdUnknown Counter32, • pcePcepSessNumReqRcvdUnknown Counter32	
ESO Consortium MIB, which can be found at http://www.snmp.com/eso/ **NOTE**: The ESO Consortium MIB has been replaced by RFC 3826.	No exceptions	ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, and T Series
Internet Assigned Numbers Authority, *IANAiftype Textual Convention MIB*	No exceptions	ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, and T Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
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	M Series, MX Series, and T Series
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.)	ACX Series, EX Series, M Series, MX Series, SRX Series, and T Series
Internet draft draft-ietf-idmr-igmp-mib-13.txt, *Internet Group Management Protocol (IGMP) MIB*	No exceptions	EX Series, M Series, MX Series, PTX Series, SRX Series, and T Series
Internet draft draft-ietf-idmr-pim-mib-09.txt, *Protocol Independent Multicast (PIM) MIB*	No exceptions	ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series , and T Series
Internet draft draft-ietf-isis-wg-mib-07.txt, *Management Information Base for IS-IS* **NOTE**: Replaced with RFC 4444, *IS-IS MIB* in Junos OS Release 11.3 and later.	Unsupported tables and objects: • isisISAdjTable • isisISAdjAreaAddrTable • isisISAdjIPAddrTable • isisISAdjProtSuppTable)	ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, and T Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
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.)	M Series, MX Series, and T Series
Internet draft draft-ietf-mpls-mldp-mib-02.txt, *Definitions of Managed Objects for the LDP Point-to-Multipoint and Multipoint-to-Multipoint Label Switched Paths*	No exceptions	M Series, MX Series, PTX Series, and T Series
Internet draft draft-ietf-mpls-p2mp-te-mib-09.txt, *P2MP MPLS-TE MIB* (read-only access)	Unsupported table: • `mplsTeP2mpTunnelBranchPerfTable`	ACX Series, M Series, MX Series, PTX Series, and T Series
Internet draft draft-ietf-ospf-ospfv3-mib-11.txt, *Management Information Base for OSPFv3*	Support for `ospfv3NbrTable` only.	M Series, MX Series, PTX Series, SRX Series, and T Series
Internet draft draft-ietf-ppvpn-mpls-vpn-mib-04.txt, *MPLS/BGP Virtual Private Network Management Information Base Using SMIv2*	Supported tables and objects: • `mplsVpnScalars` • `mplsVpnVrfTable` • `mplsVpnPerTable` • `mplsVpnVrfRouteTargetTable`	M Series, MX Series, PTX Series, and T Series

Table 54: Standard MIBs supported by Junos OS *(Continued)*

Standard MIB	Supported and unsupported tables and objects	Platforms
Internet draft draft-kamarthy-gdoi-mib-01, *Management Information Base for the Group Domain of Interpretation (GDOI)*	Caveats: • The GDOI MIB from the IETF draft is modified to include only the group member tables and notifications. • Only the SNMP notifications that are applicable to MX Series group members are supported.	MX Series
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*	No exceptions	ACX Series, EX Series, M Series, MX Series, PTX Series, SRX Series, and T Series

For information about standard SNMP MIB objects, see the SNMP MIB Explorer.

For information about RFCs, see the Standards Reference Guide.

Enterprise-Specific MIBs for Junos OS Evolved

Starting in Junos OS Evolved Release 19.1R1, the enterprise-specific MIBs listed in Table 55 on page 610 are supported. For information about enterprise-specific SNMP MIB objects, see the SNMP MIB Explorer.

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
BGP4 V2 MIB	Provides support for monitoring 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*.	No exceptions	PTX10003 and PTX10001-36 MR

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
Chassis MIBs	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 Switch Board (SSB), Switching and Forwarding Module (SFM), Switch Fabric Board (SFB), Flexible PIC Concentrators (FPCs), and PICs.	Supported traps: • jnxFruInsertion • jnxFruRemoval • jnxFruPowerOn • jnxFruPowerOff • jnxFruOnline • jnxFruOffline • jnxFruFailed • jnxFruOK • jnxPowerSupplyFailure • jnxPowerSupplyOK • jnxPowerSupplyInputFailure • jnxPowerSupplyInputOK • jnxFanFailure • jnxFanOK • jnxOverTemperature • jnxTemperatureOK Supported tables and objects: • jnxBoxClass • jnxBoxDescr • jnxBoxSerialNo • jnxBoxRevision	PTX10003 and PTX10001-36 MR

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
		• jnxBoxInstalled • jnxContentsLastChange • jnxContainersTable • jnxOperatingTable • jnxRedundancyTable • jnxContentsTable • jnxFilledTable • jnxFruTable	
Class-of-Service MIB	Provides support for monitoring interface output queue statistics per interface and per forwarding class. Provides support for monitoring Priority-Based Flow Control (PFC) statistics. The entries in the `jnxCosPfcPriorityTable` of Class-of-Service MIB include `jnxCosPfcPriorityEntry`, `jnxCosIfIndex`, `jnxCosPfcPriorityIndex`, `jnxCosPfcPriorityRequestsTx`, and `jnxCosPfcPriorityRequestsRx`.	No exceptions	PTX Series and QFX Series

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
Destination class usage (DCU) MIB	Provides support for monitoring SCU and DCU counters.	No exceptions	PTX10001-36MR, PTX10004, PTX10008, and PTX10016
DHCP	Provides SNMP support (get only) for DHCP stateless relay configurations. Stateless relay does not include support for bindings and leases tables. Supported tables and objects: • jnxJdhcpRelayStatistics • jnxJdhcpRelayIfcStats	Support does not include the following MIB objects: • jnxJdhcpLocalServerObjects • jnxJdhcpRelayBindings • jnxJdhcpRelayTraps • jnxJdhcpRelayTrapVars	PTX10001-36MR, PTX10004, PTX10008, PTX10016, QFX5130, QFX5220
DHCPv6	Provides SNMP support (get only) for DHCPv6 stateless relay configurations. Stateless relay does not include support for bindings and leases tables. Supported tables and objects: • jnxJdhcpv6RelayStatistics • jnxJdhcpv6RelayIfcStats	Support does not include the following MIB object: • jnxJdhcpv6LocalServerObjects	PTX10001-36MR, PTX10004, PTX10008, PTX10016, QFX5130, QFX5220

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
Firewall MIB	Provides bytes and packets count of interface attached policers.	Supported tables and objects: • jnxFWCntrXTable • jnxFWCntrPolicerOutSpecPktCount • jnxFWCntrPolicerOutSpecByteCount The values of the following objects in jnxFWCntrPolicerOutSpecPktCount and jnxFWCntrPolicerOutSpecByteCount are supported, whereas the rest of the MIBs are not supported and will always be zero.	PTX10001-36MR, PTX10003, PTX10004, and PTX10008

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
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. Mounts are read in on each node in the system and compiled into a list.	Supported tables and objects: • hrStorageTable • jnxHrStorage • hrSWInstalledTable • hrSystemUptime • hrSystemDate • hrSystemInitialLoadDevice • hrSystemInitialLoadParameters • hrSystemNumUsers • hrSystemProcesses • hrSystemMaxProcesses • hrMemorySize • hrSWInstalledLastChange • hrSWInstalledLastUpdateTime	PTX10003
Interface MIB	Extends the standard ifTable (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information in the `ifJnxTable` for ECN marked packets and ingress input queue drop counters.	No exceptions	PTX10003 and QFX5220

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
IPv4 MIB	Provides additional IPv4 address information, supporting the assignment of identical IPv4 addresses to separate interfaces.	No exceptions	PTX10003
IPv6 and ICMPv6 MIB	Provides IPv6 and Internet Control Message Protocol version 6 (ICMPv6) statistics.	Unsupported objects • jnxIcmpv6GlobalStats branch and the objects under it	PTX10003
LDP MIB	Provides LDP statistics and defines LDP label-switched path (LSP) notifications. LDP traps support only IPv4 standards.	No exceptions	PTX10003
MPLS LDP MIB	Contains object definitions as described in RFC 3815, *Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP)*.	No exceptions	PTX10003
MPLS MIB	Provides MPLS information and defines MPLS notifications.	No exceptions	PTX10003

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
RSVP MIB	Provides information about RSVP-traffic engineering sessions that correspond to MPLS LSPs on transit routers in the service provider core network.	No exceptions	PTX10003
SFF Digital Optical Monitor MIB	Defines objects used for Digital Optical Monitor on interfaces of Juniper products.	Supported tables: • jnxDomCurrentTable • jnxDomModuleLaneTable	PTX10003
SNMP USM HMAC-SHA-2 MIB	Contains Juniper Networks' implementation of enterprise specific MIB for SNMP USM HMAC-SHA-2.	Supported Objects: • usmHMAC128SHA224AuthProtocol • usmHMAC192SHA256AuthProtocol	ACX7100 - 32C, ACX7100 - 48L, ACX7509, ACX7900, ACX7024, PTX1000136 MR, PTX10003, PTX10004, PTX10008, PTX10016, QFX5130 - 32CD, QFX5130 - 48C, QFX5130 - 48CM, QFX5700, QFX5220, QFX5230 - 64CD

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
Source class usage (SCU) MIB	Provides support for monitoring SCU and DCU counters.	No exceptions	PTX10001-36MR, PTX10004, PTX10008, and PTX10016

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
TWAMP MIB (jnxTwampMib)	Monitors network performance using Two-Way Active Measurement Protocol.	Supported tables: • jnxTwampClientResultsSampleTable • jnxTwampClientResultsSummaryTable • jnxTwampClientResultsCalculatedTable • jnxTwampClientHistorySampleTable • jnxTwampClientHistorySummaryTable • jnxTwampClientHistoryCalculatedTable • jnxTwampClientControlConnectionTable • jnxTwampClientTestSessionsTable Supported traps: • jnxTwampClientControlConnectionClosed • jnxTwampClientTestIterationFinished • pingProbeFailed • pingTestFailed • pingTestCompleted • jnxPingRttThresholdExceeded	PTX10001-36MR, PTX10003, PTX10004, and PTX10008

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
		• jnxPingRttJitterThresholdExceeded • jnxPingEgressThresholdExceeded • jnxPingEgressJitterThresholdExceeded • jnxPingIngressThresholdExceeded • jnxPingIngressJitterThresholdExceeded • jnxPingMaxRttThresholdExceeded	

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
Timing MIB (jnxTimingNotfnsMIB)	Defines Synchronous Ethernet (SyncE) objects, faults, and events.	Supported traps: • jnxTimingFaultLOSSet • jnxTimingFaultLOSClear • jnxTimingFaultEFDSet • jnxTimingFaultEFDClear • jnxTimingFaultLOESMCSet • jnxTimingFaultLOESMCClear • jnxTimingFaultQLFailSet • jnxTimingFaultQLFailClear • jnxTimingFaultLTISet • jnxTimingFaultLTIClear • jnxTimingFaultPriSrcFailed • jnxTimingFaultSecSrcFailed • jnxTimingEventPriSrcRecovered • jnxTimingEventSecSrcRecovered • jnxTimingEventPriRefChanged • jnxTimingEventSecRefChanged • jnxTimingEventQLChangedRx • jnxTimingEventQLChangedTx • jnxTimingEventDpllStatus • jnxTimingEventSynceDpllStatus Supported objects and tables:	PTX10008

Table 55: Enterprise-Specific MIBs Supported by Junos OS Evolved *(Continued)*

Enterprise-Specific MIB	Description	Supported and unsupported tables and objects	Platform
		• jnxClksyncIfIndex • jnxClksyncIntfName • jnxClksyncQualityCode • jnxClksyncQualityCodeStr • jnxClksyncDpllState • jnxClksyncDpllStateStr • jnxClksyncSynceLockedIfIndex • jnxClksyncSynceLockedIntfName • jnxClksyncSynceQualityTable	
VPN MIB	Provides monitoring for Layer 3 VPNs, Layer 2 VPNs, and virtual private LAN service (VPLS).	Unsupported objects • jnxVpnActiveVpns • jnxVpnConfiguredVpns	PTX10003

Enterprise-Specific SNMP MIBs Supported by Junos OS

Junos OS supports the enterprise-specific MIBs listed in Table 56 on page 623. For information about enterprise-specific SNMP MIB objects, see the SNMP MIB Explorer.

Table 56: Enterprise-specific MIBs supported by Junos OS

Enterprise-Specific MIB	Description	Platforms
AAA Objects MIB	Provides support for monitoring user authentication, authorization, and accounting through the RADIUS, LDAP, SecurID, and local authentication servers.	SRX Series and vSRX Virtual Firewall
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.	SRX Series and vSRX Virtual Firewall
Alarm MIB	Provides information about alarms from the router chassis.	All platforms except MX10003 and MX204 devices.
Analyzer MIB	Provides information about analyzer and remote analyzer related to port mirroring on the EX Series Ethernet Switches.	EX Series, QFabric system, and QFX Series
Antivirus Objects MIB	Provides information about the antivirus engine, antivirus scans, and antivirus scan-related traps.	SRX Series and vSRX Virtual Firewall
ATM Class-of-Service MIB	Provides support for ATM interfaces and virtual connections.	ACX Series, M Series, and T Series
ATM 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.	M Series, SRX Series, T Series and vSRX Virtual Firewall

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
BGP4 V2 MIB	Provides support for monitoring 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*.	All platforms
BGP MIB	Contains the objects for BGP version.	MX Series
Bidirectional Forwarding Detection MIB	Provides support for monitoring Bidirectional Forwarding Detection (BFD) sessions.	All platforms
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.	SRX Series and vSRX Virtual Firewall
Chassis Definitions for Router Model MIB	Contains the object identifiers (OIDs) that are used by the Chassis MIB to identify platform and chassis components. The Chassis MIB provides information that changes often, whereas the Chassis Definitions for Router Model MIB provides information that changes less often.	ACX Series, M Series, MX Series, PTX Series, QFX Series, SRX550, SRX1500, and T Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Chassis MIBs	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 Switch Board (SSB), Switching and Forwarding Module (SFM), Switch Fabric Board (SFB), Flexible PIC Concentrators (FPCs), and PICs.	All platforms
Class-of-Service MIB	Provides support for monitoring interface output queue statistics per interface and per forwarding class. Provides support for monitoring Priority-Based Flow Control (PFC) statistics. The entries in the `jnxCosPfcPriorityTable` of Class-of-Service MIB include `jnxCosPfcPriorityEntry`, `jnxCosIfIndex`, `jnxCosPfcPriorityIndex`, `jnxCosPfcPriorityRequestsTx`, and `jnxCosPfcPriorityRequestsRx`.	ACX Series, EX Series, M Series, MX Series, PTX Series, QFabric system, QFX Series, SRX Series, T Series, and vSRX Virtual Firewall
CGNAT MIB	Provides information about services interfaces used for CGNAT implementation. • SRX – USF (MX-SPC3) JUNIPER-JS-NAT-MIB • MS-MPC JUNIPER-NET-MIB	MX Series and SRX Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
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. History of the last 32 configuration changes is kept in `jnxCmChgEventTable`.	All platforms
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 the 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.	EX Series, M Series, SRX Series, T Series, and vSRX Virtual Firewall
DHCP 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.	M Series, MX Series, and T Series
DHCPv6 MIB	Provides SNMP support (get and trap) for DHCPv6 local server and relay configurations. It also provides support for bindings and leases tables, and for statistics.	M Series, MX Series, and T Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Digital Optical Monitoring MIB	Provides support for the **SNMP Get** request for statistics and **SNMP Trap** notifications for alarms.	EX Series, M Series, MX Series, PTX Series, and T Series
DNS Objects MIB	Provides support for monitoring DNS proxy queries, requests, responses, and failures.	SRX Series and vSRX Virtual Firewall
Dynamic Flow Capture MIB	Provides support for monitoring the operational status of dynamic flow capture (DFC) PICs.	M Series and T Series
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.	EX Series, M Series, MX Series, QFX Series, SRX300, SRX320, SRX340, SRX550, SRX1500 and T Series
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.	ACX Series, EX Series, M Series, MX Series, PTX Series, QFabric system, QFX Series, SRX1500, SRX300, SRX320, SRX340, SRX550, and T Series
Experimental MIB	Contains object identifiers for experimental MIBs.	ACX Series, M series, MX Series, and T series
EX Series MAC Notification MIB	Contains Juniper Networks' implementation of enterprise-specific MIB for Ethernet Mac Stats for EX Series.	EX Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
EX Series SMI MIB	Contains the Structure of Management Information for Juniper Networks EX Series platforms.	EX Series
Firewall MIB	Provides support for monitoring *firewall filter* counters. Routers must have the Internet Processor II ASIC to perform firewall monitoring.	ACX Series, EX Series, M Series, MX Series, PTX Series, QFabric system, QFX Series, SRX300, SRX320, SRX340, SRX550, SRX1500 and T Series
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.	M Series and T Series
GRE Keepalive Monitoring MIB	Provides support for monitoring generic routing encapsulation (GRE) keepalive status. This MIB also provides an SNMP trap when GRE keepalive status changes.	SRX Series and vSRX Virtual Firewall instances
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` measure the usage in allocation units—`hrStorageUsed` and `hrStorageAllocationUnits`—only. Using the percentage measurement, you can monitor and apply thresholds on usage.	ACX Series, EX Series, M Series, MX Series, QFX Series, SRX300, SRX320, SRX340, SRX550, SRX1500 and T Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Interface Accounting Forwarding Class MIB	Extends the Juniper Enterprise Interface MIB and provides support for monitoring statistics data for interface accounting and IETF standardization.	M Series, MX Series, SRX Series, and vSRX Virtual Firewall
Interface MIB	Extends the standard `ifTable` (RFC 2863) with additional statistics and Juniper Networks enterprise-specific chassis information.	ACX Series, EX Series, M Series, MX Series, PTX Series, QFabric system, QFX Series, SRX300, SRX320, SRX340, SRX550, SRX1500 and T Series
IP Forward MIB	Extends the standard IP Forwarding Table MIB (RFC 4292) to include CIDR forwarding information.	All platforms
IPsec Generic Flow Monitoring Object MIB	Based on `jnx-ipsec-monitor-mib`, this MIB provides support for monitoring IPsec and IPsec VPN management objects.	SRX Series and vSRX Virtual Firewall
IPsec Monitoring MIB	Provides operational and statistical information related to the IPsec and IKE tunnels on Juniper Networks routers.	M Series, SRX Series, and T Series
IPsec VPN Objects MIB	Provides support for monitoring IPsec and IPsec VPN management objects for Juniper products. This MIB is an extension of `jnx-ipsec-flow-mon.mib`.	SRX Series and MX Series with USF
IPv4 MIB	Provides additional Internet Protocol version 4 (IPv4) address information, supporting the assignment of identical IPv4 addresses to separate interfaces.	All platforms

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
IPv6 and ICMPv6 MIB	Provides IPv6 and Internet Control Message Protocol version 6 (ICMPv6) statistics.	M series, MX Series, PTX Series, SRX Series, T Series, and vSRX Virtual Firewall
jnxASICExternalMemTraps	Provides information on ASIC external memory error.	QFX10002-36Q, QFX10002-60C, QFX10002-72Q, QFX10008, QFX10016, PTX1000, PTX10002-60C, PTX10008, PTX10016
jnxASICExternalMemOKTraps	Provides information on ASIC external memory error.	QFX10002-36Q, QFX10002-60C, QFX10002-72Q, QFX10008, QFX10016, PTX1000, PTX10002-60C, PTX10008, PTX10016
jnxHmcFatal	Provides information when the specified HMC on a specific FPC has failed.	QFX10002-36Q, QFX10002-60C, QFX10002-72Q, QFX10008, QFX10016, PTX1000, PTX10002-60C, PTX10008, PTX10016
jnxHmcOK	Provides information when the specified HMC on a specific FPC has recovered from the failure.	QFX10002-36Q, QFX10002-60C, QFX10002-72Q, QFX10008, QFX10016, PTX1000, PTX10002-60C, PTX10008, PTX10016
jnxJsChassisHA	Provides Chassis High Availability with ensuring minimal disruption to services in case of a failover. If one of the chassis in a High Availability environment fails, the other chassis takes over the function of the failed chassis with minimal service interruption. This module defines the objects pertaining to Chassis High Availability.	SRX5400, SRX5600, and SRX5800.

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
jnxJsFlowSofSummary MIB	Provides the total number of Express Path mode (formerly known as services offloading) sessions in use and total number of packets processed so far in logical system.	SRX4600, SRX5400, SRX5600, and SRX5800.
jnxJsChNodeCPUStatus	Monitors Routing Engine CPU load usage. It sends a notification to users when Routing Engine CPU load is below set threshold.	SRX5400, SRX5600, SRX5800, SRX4600, SRX4200, SRX4100, and SRX1500.
jnxJsChNodeJunosKernelStatus	Monitors Junos Kernel usage.	SRX5400, SRX5600, SRX5800, SRX4600, SRX4200, SRX4100, and SRX1500.
jnxUserFirewalls MIB	Exports statistics of User Firewall identity-management counters.	SRX Series and vSRX Virtual Firewall
jnxTLBMIB	Exports statistics of Traffic Load Balancer application	MX240, MX480, and MX960
JNX BGP MIB2	Support IPV6 objects and prefix counters for BGP.	MX Series
JNX VPN MIB (L2VPN)	Contains information about L2VPN protocol.	MX Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
L2ALD MIB	Contains information about the Layer 2 Address Learning process (L2ALD) and related traps, such as the routing instance MAC limit trap and the interface MAC limit trap. This MIB also provides VLAN information in the `jnxL2aldVlanTable` table for Enhanced Layer 2 Software (ELS) EX Series and QFX Series switches. **NOTE**: Non-ELS EX Series switches support the VLAN MIB (`jnxExVlanTable` table) for VLAN information instead of this MIB. .	EX Series, MX Series, QFX Series, and T Series
L2CP MIB	Provides information about Layer 2 Control Protocols (L2CP) based features. Currently, Junos OS supports only the `jnxDot1dStpPortRootProtectEnabled`, `jnxDot1dStpPortRootProtectState`, and `jnxPortRootProtectStateChangeTrap` objects.	MX Series
L2TP MIB	Provides information about Layer 2 Transport Protocol (L2TP) tunnels and sessions.	M Series, MX Series, and T Series
LDP MIB	Provides LDP statistics and defines LDP label-switched path (LSP) notifications. LDP traps support only IPv4 standards.	ACX Series, M Series, PTX Series, SRX Series, and T Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
License MIB	Extends SNMP support to licensing information, and introduces SNMP traps that alert users when the licenses are about to expire, expired, or when the total number of users exceeds the number specified in the license.	M Series, MX Series, SRX Series, and T Series
Logical Systems MIB	Extend SNMP support to logical systems security profile through various MIBs defined under `jnxLsysSecurityProfile`.	SRX Series
LTE MIB	Extend SNMP support to monitor the 4G LTE Mini-Physical Interface Module (Mini-PIM) status using SNMP remote network management.	SRX300, SRX320, SRX340, SRX345, and SRX550M.
LSYSTSYS MIB (jnxLsysVD)	Provides the following details of configured logical systems and tenant: • total LSYS count • total TSYS count • total security profiles count • maximally allowed LSYS capacity • maximally allowed TSYS capacity • maximally allowed security profiles capacity	SRX1500, SRX4100, SRX4200, SRX4600, SRX5400, SRX5600, SRX5800, and vSRX Virtual Firewall

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
MIMSTP MIB	Provides information about MSTP instances (that is, routing instances of type Virtual Switch/Layer 2 control, also known as virtual contexts), MSTIs within the MSTP instance, and VLANs associated with the MSTI.	MX Series and T Series
MPLS LDP MIB	Contains object definitions as described in RFC 3815, *Definitions of Managed Objects for the Multiprotocol Label Switching (MPLS), Label Distribution Protocol (LDP)*. **NOTE**: Objects in the MPLS LDP MIB are supported in earlier releases of Junos OS as a proprietary LDP MIB (`mib-ldpmib.txt`). As the branch used by the proprietary LDP (`mib-ldpmib.txt`) conflicts with RFC 3812, the proprietary LDP MIB (`mib-ldpmib.txt`) has been deprecated and replaced by the enterprise-specific MPLS LDP MIB (`mib-jnx-mpls-ldp.txt`).	ACX Series, EX Series, M Series, MX Series, PTX Series, QFX Series, and T Series
MPLS MIB	Provides MPLS information and defines MPLS notifications. **NOTE**: To collect information about MPLS statistics on transit routers, use the enterprise-specific RSVP MIB (`mib-jnx-rsvp.txt`) instead of the enterprise-specific MPLS MIB (`mib-jnx-mpls.txt`).	ACX Series, EX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, and T Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
MVPN MIB	Contains objects that enable SNMP manager to monitor MVPN connections on the provider edge routers. The enterprise-specific MVPN MIB is the Juniper Networks extension of the IETF standard MIBs defined in Internet draft draft-ietf-l3vpn-mvpn-mib-03.txt, *MPLS/BGP Layer 3 VPN Multicast Management Information Base*.	All platforms
MPLS L3VPN MIB	Contains the attributes for L3VPN based MPLS.	MX Series
MPLS VPN MIB	Contains the objects for MPLS VPN.	MX Series
NAT Objects MIB	Provides support for monitoring network address translation (NAT).	EX Series and SRX Series
NAT Resources-Monitoring MIB	Provides support for monitoring NAT pools usage and NAT rules. Notifications of usage of NAT resources are also provided by this MIB. This MIB is currently supported on the Multiservices PIC and Multiservices DPC on M Series and MX Series routers only.	M Series and MX Series
OTN Interface Management MIB	Defines objects for managing Optical Transport Network (OTN) interfaces on devices running Junos OS.	M Series, MX series, PTX Series, and T Series
Packet Forwarding Engine MIB	Provides notification statistics for Packet Forwarding Engines.	ACX Series, EX Series, M Series, PTX Series, SRX Series, and T Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Packet Mirror MIB	Enables you to capture and view packet mirroring-related information. This MIB is currently supported by Junos OS for MX Series routers only. Packet mirroring traps are an extension of the standard SNMP implementation and are only available to SNMPv3 users.	MX Series
PAE Extension MIB	Extends the standard IEEE802.1x PAE Extension MIB, and contains information for Static MAC Authentication.	EX Series
Passive Monitoring MIB	Performs traffic flow monitoring and lawful interception of packets transiting between two routers.	M Series and T Series
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.	ACX Series, EX Series, M Series, MX Series, QFX Series, SRX Series, and T Series
Policy Objects MIB	Provides support for monitoring the security policies that control the flow of traffic from one zone to another.	SRX Series
Power Supply Unit MIB	Enables monitoring and managing of the power supply on a device running Junos OS.	EX Series and QFabric system

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
PPP MIB	Provides SNMP support for PPP-related information such as the type of authentication used, interface characteristics, status, and statistics. This MIB is supported on Common Edge PPP process, jpppd.	M Series and MX Series
PPPoE MIB	Provides SNMP support for PPPoE-related information such as the type of authentication used, interface characteristics, status, and statistics. This MIB is supported on Common Edge PPPoE process, jpppoed.	M Series and MX Series
Pseudowire ATM MIB	Extends the standard Pseudowire MIB, and defines objects used for managing the ATM pseudowires in Juniper products. The enterprise-specific Pseudowire ATM MIB is the Juniper Networks implementation of RFC 5605, *Managed Objects for ATM over Packet Switched Networks (PSNs)*.	M Series and MX Series
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).	ACX Series, M Series, and T Series
PTP MIB	Monitors the operation of PTP clocks within the network.	MX Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Real-Time Performance Monitoring MIB	Provides real-time performance-related data and enables you to access *jitter* measurements and calculations using SNMP.	EX Series, M Series, MX Series, SRX Series, and T Series
Reverse-Path-Forwarding MIB	Monitors statistics for traffic that is rejected because of reverse-path-forwarding (RPF) processing.	All platforms
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.	All platforms
RSVP MIB	Provides information about RSVP-traffic engineering sessions that correspond to MPLS LSPs on transit routers in the service provider core network. **NOTE**: To collect information about MPLS statistics on transit routers, use the enterprise-specific RSVP MIB (`mib-jnx-rsvp.txt`) instead of the enterprise-specific MPLS MIB (`mib-jnx-mpls.txt`).	ACX Series, M Series, MX Series, PTX Series, and T Series
Service OAM MIB	The `jnx-soam-pm.mib` MIB provides SNMP support for service OAM performance monitoring functions.	SRX380, SRX300, SRX320, SRX340, SRX345, and MX Series.
Security Interface Extension Objects MIB	Provides support for the security management of interfaces.	EX Series, SRX Series, and vSRX Virtual Firewall

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Security Screening Objects MIB	Defines the MIB for the Juniper Networks Enterprise Firewall screen functionality.	SRX Series and vSRX Virtual Firewall
Services PIC MIB	Provides statistics for Adaptive Services (AS) PICs and defines notifications for AS PICs.	M Series and T Series
SNMP IDP MIB	Contains Juniper Networks' implementation of enterprise specific MIB for IDP.	SRX Series and vSRX Virtual Firewall
SONET APS MIB	Monitors any SONET interface that participates in Automatic Protection Switching (APS).	M Series and T Series
SONET/SDH Interface Management MIB	Monitors the current alarm for each SONET/SDH interface.	M Series and T Series
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.	M Series, T Series, and SRX Series
SPU Monitoring MIB	Provides support for monitoring SPUs on SRX5600 and SRX5800 devices.	SRX Series and vSRX Virtual Firewall

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Structure of Management Information MIB	Explains how the Juniper Networks enterprise-specific MIBs are structured.	ACX Series, EX Series, M Series, MX series, QFX Series, SRX Series, T Series and vSRX Virtual Firewall
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.	EX Series
Structure of Management Information MIB for SRX Series	Contains object identifiers (OIDs) for the security branch of the MIBs used in Junos OS for SRX Series Firewalls, services, and traps.	SRX Series and vSRX Virtual Firewall
Subscriber MIB	Provides SNMP support for subscriber-related information.	ACX Series, MX Series, and T Series
System Log MIB	Enables notification of an SNMP trap-based application when an important system log message occurs.	EX Series, M Series, MX Series, PTX Series, QFX Series, SRX Series, and T Series
Timing MIB	Defines Synchronous Ethernet (SyncE) and Precision Time Protocol (PTP) objects, faults, and events.	ACX710
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.	EX Series, M Series, MX Series, SRX Series, T Series, and vSRX Virtual Firewall

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
Tunnel Stats MIB	Supports monitoring of tunnel statistics for IPV4 over IPV6 tunnels. This MIB currently displays three counters: tunnel count in rpd, tunnel count in Kernel, and tunnel count in the Packet Forwarding Engine.	all platforms
Utility MIB	Provides SNMP support for exposing the Junos OS data and has tables that contain information about each type of data, such as integer and string.	EX Series, M Series, MX Series, QFabric system, QFX Series, SRX Series, T Series, and vSRX Virtual Firewall
Virtual Chassis MIB	Contains information about the virtual chassis on the EX Series Ethernet Switches and the MX Series.	EX Series and MX Series
VLAN MIB	Contains information about prestandard IEEE 802.10 VLANs and their association with LAN emulation clients. **NOTE**: For ELS EX Series switches and QFX Series switches, VLAN information is provided in the L2ALD MIB in the `jnxL2aldVlanTable` table instead of in this MIB. Non-ELS EX Series Ethernet switches use the `jnxExVlanTable` table in this MIB to provide VLAN configuration information, and the `jnxVlanTable` table in this MIB has been deprecated and is no longer used.	EX Series and QFX Series

Table 56: Enterprise-specific MIBs supported by Junos OS *(Continued)*

Enterprise-Specific MIB	Description	Platforms
VPLS MIBs	Provides information about generic, BGP-based, and LDP-based VPLS, and pseudowires associated with the VPLS networks. The enterprise-specific VPLS MIBs are Juniper Networks extensions of the following IETF standard MIBs defined in Internet draft draft-ietf-l2vpn-vpls-mib-05.txt, and are implemented as part of the `jnxExperiment` branch: • `VPLS-Generic-Draft-01-MIB` implemented as `mib-jnx-vpls-generic.txt` • `VPLS-BGP-Draft-01-MIB` implemented as `mib-jnx-vpls-bgp.txt` • `VPLS-LDP-Draft-01-MIB` implemented as `mib-jnx-vpls-ldp.txt`	M Series, MX Series, and T Series
VPN Certificate Objects MIB	Provides support for monitoring the local and CA certificates loaded on the router.	EX Series, SRX Series, and vSRX Virtual Firewall
VPN MIB	Provides monitoring for Layer 3 VPNs, Layer 2 VPNs, and virtual private LAN service (VPLS) (read access only).	ACX Series, EX Series, M Series, MX Series, and T Series

Starting in Junos OS Release 18.4R1, you can monitor 4G LTE Mini-PIM status by using SNMP remote network management.

You can use the following commands to monitor the 4G LTE Mini-PIM status:

```
show snmp mib walk ascii jnxWirelessWANNetworkInfoTable

show snmp mib walk ascii jnxWirelessWANFirmwareInfoTable
```

Starting in Junos OS Release 19.4R1, on SRX5000 line devices with SRX5K-SPC3 card, we have enhanced the existing IPsec VPN flow monitor MIB **jnxIpSecFlowMonMIB** to support the global IKE statistics for tunnels using IKEv2. Use the `show security ike stats` command to display the global statistics of tunnels such as in-progress, established, and expired negotiations using IKEv2.

Starting in Junos OS Release 20.1R1, you can enable the peer down and IPsec tunnel down traps and configure the certificate authority (CA) and local certificate traps. We've enhanced the existing IPsec VPN flow monitor MIB **jnxIpSecFlowMonMIB** to support the global data plane, active IKE SA, active IPsec SA, and active peer statistics for tunnels using IKEv2. We've also enhanced the output of the `show security ike stats` command to add additional options (`<brief> | <detail>`). Use the `clear security ike stats` command to clear the IKEv2 statistic counters.

Starting in Junos OS Release 20.4R1, you can monitor CPU and Kernel usage on Routing Engine using reswatch process.

SEE ALSO

Network Management and Monitoring Guide

Standard MIBs for Junos OS Evolved

Table 57 on page 643 shows the Standard MIBs supported in Junos OS Evolved. For information about Standard MIB objects, see the SNMP MIB Explorer.

Table 57: Standard MIBs Supported by Junos OS Evolved

Standard MIB	Exceptions	Platforms
RFC 1155, *Structure and Identification of Management Information for TCP/IP-Based Internets*	No exceptions	PTX10003
RFC 1157, *A Simple Network Management Protocol (SNMP)*	No exceptions	PTX10003
RFC 1212, *Concise MIB Definitions*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 1213, *Management Information Base for Network Management of TCP/IP-Based Internets: MIB-II*	Unsupported tables and objects: • ICMP group	PTX10003
RFC 1215, *A Convention for Defining Traps for Use with the SNMP*	No exceptions	PTX10003
RFC 1850, *OSPF Version 2 Management Information Base*	No exceptions	PTX10003
RFC 1901, *Introduction to Community-Based SNMPv2*	No exceptions	PTX10003
RFC 2011, *SNMPv2 Management Information Base for the Internet Protocol Using SMIv2*	No exceptions	PTX10003
RFC 2096, *IP Forwarding Table MIB*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 2465, *Management Information Base for IP Version 6: Textual Conventions and General Group*	Supported tables and objects: • ipv6AddrTable • ipv6NetToMediaTable • ipv6IfTable • ipv6IfStatsTable • ipv6AddrPrefixTable • ipv6IfTableLastChange • ipv6Interfaces • ipv6Forwarding • ipv6DefaultHopLimit	PTX10003
RFC 2576, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*	No exceptions	PTX10003
RFC 2578, *Structure of Management Information Version 2 (SMIv2)*	No exceptions	PTX10003
RFC 2579, *Textual Conventions for SMIv2*	No exceptions	PTX10003
RFC 2580, *Conformance Statements for SMIv2*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 2665, *Definitions of Managed Objects for the Ethernet-like Interface Types*	Unsupported tables and objects: • dot3	PTX10003
RFC 2790, *Host Resources MIB*	Unsupported tables and objects: • hrDeviceTable • hrSWRunTable • hrSWRunPerfTable	PTX10003
RFC 2863, *The Interfaces Group MIB*	No exceptions	PTX10003
RFC 2864, *The Inverted Stack Table Extension to the Interfaces Group MIB*	No exceptions	PTX10003
RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*	No exceptions	PTX10003
RFC 2932, *IPv4 Multicast Routing MIB*	No exceptions	PTX10003
RFC 2934, *Protocol Independent Multicast MIB for IPv4*	No exceptions	PTX10003
RFC 2981, *Event MIB*	No exceptions	PTX10003
RFC 3014, *Notification Log MIB*	No exceptions	PTX10003
RFC 3019, *IP Version 6 Management Information Base for the Multicast Listener Discovery Protocol*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 3410, *Introduction and Applicability Statements for Internet-Standard Management Framework*	No exceptions	PTX10003
RFC 3411, *An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*	No exceptions	PTX10003
RFC 3412, *Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)*	No exceptions	PTX10003
RFC 3413, *Simple Network Management Protocol (SNMP) Applications*	No exceptions	PTX10003
RFC 3414, *User-Based Security Model (USM) for Version 3 of the Simple Network Management Protocol (SNMPv3)*	No exceptions	PTX10003
RFC 3415, *View-Based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)*	No exceptions	PTX10003
RFC 3416, *Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)*	No exceptions	PTX10003
RFC 3417, *Transport Mappings for the Simple Network Management Protocol (SNMP)*	No exceptions	PTX10003
RFC 3418, *Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)*	No exceptions	PTX10003
RFC 3584, *Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework*	No exceptions	PTX10003
RFC 3635, *Definitions of Managed Objects for the Ethernet-Like Interface Types*	No exceptions	PTX10003, PTX10008

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 3637, *Definitions of Managed Objects for the Ethernet WAN Interface Sublayer*	No exceptions	PTX10003
RFC 3811, *Definitions of Textual Conventions (TCs) for Multiprotocol Label Switching (MPLS) Management*	No exceptions	PTX10003
RFC 3812, *Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)* (read-only access)	No exceptions	PTX10003
RFC 3813, *Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information Base (MIB)*	Unsupported tables and objects (read only access): • mplsInterfacePerfTable • mplsInSegmentPerfTable • mplsOutSegmentPerfTable • mplsInSegmentMapTable • mplsXCUp • mplsXCDown	PTX10003
RFC 3826, *The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-Based Security Model*	No exceptions	PTX10003
RFC 3877, *Alarm Management Information Base*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
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.	PTX Series (PTX10008, PTX10001-36MR, PTX10001, and PTX10004)
RFC 4133, Entity MIB	Supported table: • entPhysicalTable • entPhysicalModelName—Provides information for FRU (field replaceable units) inventory and health check using SNMP.	PTX10003
RFC 4292, *IP Forwarding MIB*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 4293, Management Information Base for the Internet Protocol (IP)	Supported tables: • ipAddressTable • ipAddrTable • ipNetToPhysicalTable • ipNetToMediaTable • ipSystemStatsTable Unsupported objects: • icmpMsgStatsIPVersion • icmpMsgStatsType • icmpMsgStatsInPkts • icmpMsgStatsOutPkts • icmpStatsIPVersion • icmpStatsInMsgs • icmpStatsInErrors • icmpStatsOutMsgs • icmpStatsOutErrors	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
RFC 4293, Management Information Base for the Internet Protocol (IP)	Supported tables: • icmpStatsTable • icmpMsgStatsTable	ACX7100-32C, PTX10008, and QFX10008
RFC 4444, *IS-IS MIB*	No exceptions	PTX10003
RFC 5643, *Management Information Base for OSPFv3* (read-only access)	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
IEEE, 802.3ad, *Aggregation of Multiple Link Segments*	Supported objects for PTX10008 on Junos OS Evolved Release 20.1R1: • dot3adAggPortStatsLACPDUsRx, dot3adAggPortStatsMarkerPDUsRx, dot3adAggPortStatsMarkerResponsePDUsRx, dot3adAggPortStatsUnknownRx, dot3adAggPortStatsIllegalRx, dot3adAggPortStatsLACPDUsTx, dot3adAggPortStatsMarkerPDUsTx, and dot3adAggPortStatsMarkerResponsePDUsTx • dot3adInterfaceName, dot3adOperState, dot3adAggname, and dot3adInterfaceTimeout. Unsupported objects for PTX10008 on Junos OS Evolved Release 20.1R1: • dot3adAggActorSystemPriority, dot3adAggActorSystemID,	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
	dot3adAggActorAdminKey, and dot3adAggActorOperKey. • dot3adAggMACAddress, dot3adAggAggregateOrIndividual, dot3adAggPartnerSystemID, dot3adAggPartnerSystemPriority, dot3adAggPartnerOperKey, dot3adAggCollectorMaxDelay, dot3adAggPortListPorts, and dot3adTablesLastChanged • dot3adAggPortActorSystemPriority, dot3adAggPortActorSystemID, dot3adAggPortActorAdminKey, dot3adAggPortActorOperKey, dot3adAggPortActorPort,dot3adAggPortActorPortPriority, dot3adAggPortActorAdminState, and dot3adAggPortActorOperState	

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
	• dot3adAggPortPartnerAdminSystemPriority, dot3adAggPortPartnerOperSystemPriority, dot3adAggPortPartnerAdminSystemID, dot3adAggPortPartnerOperSystemID, dot3adAggPortPartnerAdminKey,dot3adAggPortPartnerOperKey, dot3adAggPortPartnerAdminPort, dot3adAggPortPartnerOperPort, dot3adAggPortPartnerAdminPortPriority, and dot3adAggPortPartnerOperPortPriority • dot3adAggPortDebugRxState, dot3adAggPortDebugLastRxTime, dot3adAggPortDebugMuxState, dot3adAggPortDebugMuxReason, dot3adAggPortDebugActorChurnState, dot3adAggPortDebugPartnerChurnState, dot3adAggPortDebug	

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
	`ActorChurnCount`, `dot3adAggPortDebugPartnerChurnCount`, `dot3adAggPortDebugActorSyncTransitionCount`, `dot3adAggPortDebugPartnerSyncTransitionCount`, `dot3adAggPortDebugActorChangeCount`, and `dot3adAggPortDebugPartnerChangeCount`.	
Internet Assigned Numbers Authority, *IANAiftype Textual Convention MIB*	No exceptions	PTX10003
Internet draft draft-ietf-idmr-igmp-mib-13.txt, *Internet Group Management Protocol (IGMP) MIB*	No exceptions	PTX10003
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*	No exceptions	PTX10003
Internet draft draft-ietf-isis-wg-mib-07.txt, *Management Information Base for IS-IS*	No exceptions	PTX10003
Internet draft draft-ietf-ospf-ospfv3-mib-11.txt, *Management Information Base for OSPFv3*	No exceptions	PTX10003
Internet draft draft-ietf-idmr-pim-mib-09.txt, *Protocol Independent Multicast (PIM) MIB*	No exceptions	PTX10003

Table 57: Standard MIBs Supported by Junos OS Evolved *(Continued)*

Standard MIB	Exceptions	Platforms
Internet Draft P2MP MPLS-TE MIB (draft-ietf-mpls-p2mp-te-mib-09.txt) (read-only access)	No exceptions	PTX10003

Junos OS SNMP FAQs

SUMMARY

This document presents the most frequently asked questions about the features and technologies used to implement SNMP services on Juniper Networks devices using the Junos operating system.

IN THIS SECTION

- Junos OS SNMP Support FAQs | **656**
- Junos OS MIBs FAQs | **657**
- Junos OS SNMP Configuration FAQs | **666**
- SNMPv3 FAQs | **671**
- SNMP Interaction with Juniper Networks Devices FAQs | **673**
- SNMP Traps and Informs FAQs | **675**
- Junos OS Dual Routing Engine Configuration FAQs | **683**
- SNMP Support for Routing Instances FAQs | **684**
- SNMP Counters FAQs | **685**

SNMP enables users to monitor network devices from a central location.

Junos OS SNMP Support FAQs

This section provides frequently asked questions and answers related to SNMP support on Junos OS.

Which SNMP versions does Junos OS support?

Junos OS supports SNMP version 1 (SNMPv1), version 2 (SNMPv2c), and version 3 (SNMPv3). By default, SNMP is disabled on a Juniper Networks device.

Which ports (sockets) does SNMP use?

The default port for SNMP queries is port 161. The default port for SNMP traps and informs is port 162. The port used for SNMP traps and informs is configurable, and you can configure your system to use ports other than the default port 162. However, the SNMP listening port will remain the same; this is established on the RFC.

Is SNMP support different among the Junos OS platforms?

No, SNMP support is not different among the Junos OS platforms. SNMP configuration, interaction, and behavior are the same on any Junos OS device. The only difference that might occur across platforms is MIB support.

See also SNMP MIB Explorer for a list of MIBs that are supported across the Junos OS platforms.

Does Junos OS support the user-based security model (USM)?

Yes, Junos OS supports USM as part of its support for SNMPv3. SNMPv3 contains more security measures than previous versions of SNMP, including providing a defined USM. SNMPv3 USM provides message security through data integrity, data origin authentication, message replay protection, and protection against disclosure of the message payload.

Does Junos OS support the view-based access control model (VACM)?

Yes, Junos OS supports VACM as part of its support for SNMPv3. SNMPv3 contains more security measures than previous versions of SNMP, including providing a defined VACM. SNMPv3 VACM determines whether a specific type of access (read or write) to the management information is allowed.

Does Junos OS support SNMP informs?

Yes, Junos OS supports SNMP informs as part of its support for SNMPv3. SNMP informs are confirmed notifications sent from SNMP agents to SNMP managers when significant events occur on a network device. When an SNMP manager receives an inform, it sends a response to the sender to verify receipt of the inform.

Can I provision or configure a device using SNMP on Junos OS?

No, provisioning or configuring a device using SNMP is not allowed on Junos OS.

Junos OS MIBs FAQs

This section presents frequently asked questions and answers related to Junos OS MIBs.

What is a MIB?

A management information base (MIB) is a table of definitions for managed objects in a network device. MIBs are used by SNMP to maintain standard definitions of all of the components and their operating conditions within a network device. Each object in the MIB has an identifying code called an object identifier (OID).

MIBs are either standard or enterprise-specific. Standard MIBs are created by the Internet Engineering Task Force (IETF) and documented in various RFCs. Enterprise-specific MIBs are developed and supported by a specific equipment manufacturer.

For a list of supported standard MIBs, see " Standard SNMP MIBs Supported by Junos OS" on page 594.

For a list of Juniper Networks enterprise-specific MIBs, see "Enterprise-Specific SNMP MIBs Supported by Junos OS" on page 622.

Do MIB files reside on the Junos OS devices?

No, MIB files do not reside on the Junos OS devices. You must download the MIB files from the Juniper Networks Technical Publications page for the required Junos OS release: SNMP MIB Explorer.

How do I compile and load the Junos OS MIBs onto an SNMP manager or NMS?

For your network management systems (NMSs) to identify and understand the MIB objects used by 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 names, IDs, and data types for the NMS.

You can download the Junos OS MIB package from the Enterprise-Specific MIBs and Traps section at SNMP MIB Explorer or https://www.juniper.net/documentation/software/junos/index.html .

The Junos OS MIB package has two folders: `StandardMibs`, containing standard MIBs supported on Juniper Networks devices, and `JuniperMibs`, containing Juniper Networks enterprise-specific MIBs. You *must* have the required standard MIBs downloaded and decompressed before downloading any enterprise-specific MIBs. There might be dependencies that require a particular standard MIB to be present on the compiler before loading a particular enterprise-specific MIB.

The Junos OS MIB package is available in `.zip` and `.tar` formats. Download the format appropriate for your requirements.

Use the following steps to load MIB files for devices running Junos OS:

1. Navigate to the appropriate Juniper Networks software download page and locate the `Enterprise MIBs` link under the `Enterprise-Specific MIBs and Traps` section.

   NOTE: Although the link is titled `Enterprise MIBs`, both standard MIBs and enterprise-specific MIBs are available for download from this location.

2. Click the TAR or ZIP link to download the Junos OS MIB package.
3. Decompress the file (.tar or .zip) using an appropriate utility.

   NOTE: Some commonly used MIB compilers are preloaded with standard MIBs. You can skip Step 4 and Step 5 and proceed to Step 6 if you already have the standard MIBs loaded on your system.

4. Load the standard MIB files from the StandardMibs folder.

   Load the files in the following order:

   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

5. Load any 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 standard MIB to be present on the compiler before loading a particular enterprise-specific MIB. Dependencies are listed in the IMPORT section of the MIB file.

6. After loading the standard MIBs, load the Juniper Networks enterprise-specific SMI MIB, mib-jnx-smi.txt, and the following optional SMI MIBs based on your requirements:

   - mib-jnx-exp.txt—(Recommended) for Juniper Networks experimental MIB objects
   - mib-jnx-js-smi.txt—(Optional) for Juniper Security MIB tree objects

- mib-jnx-ex-smi.txt—(Optional) for EX Series Ethernet Switches

7. Load any remaining desired enterprise-specific MIBs from the `JuniperMibs` folder.

> **TIP**: While loading a MIB file, if the compiler returns an error message indicating that any of the objects are 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 are not loaded on the compiler, load the missing file or files first, then try to load the MIB file that failed.
>
> The system might return an error if files are not loaded in a particular order.

What is SMI?

Structure of Management Information Version (SMI) is a subset of Abstract Syntax Notation One (ASN.1), which describes the structure of objects. SMI is the notation syntax, or “grammar”, that is the standard for writing MIBs.

Which versions of SMI does Junos OS support?

The Junos OS supports SMIv1 for SNMPv1 MIBs, and SMIv2 for SNMPv2c and enterprise MIBs.

Does Junos OS support MIB II?

Yes, Junos OS supports MIB II, the second version of the MIB standard.

The features of MIB II include:

- Additions that reflect new operational requirements.
- Backward compatibility with the original MIBs and SNMP.
- Improved support for multiprotocol entities.
- Improved readability.

Are the same MIBs supported across all Juniper Networks devices?

There are some common MIBs supported by all the Junos OS devices, such as the Interface MIB (ifTable), System MIB, and Chassis MIB. Some MIBs are supported only by functionalities on specific platforms. For example, the Bridge MIB is supported on the EX Series Ethernet Switches and the SRX Series Firewalls for the branch.

What is the system object identifier (SYSOID) of a device? How do I determine the SYSOID of my device?

The jnx-chas-defines (Chassis Definitions for Router Model) MIB has a `jnxProductName` branch for every Junos OS device. The system object ID of a device is identical to the object ID of the `jnxProductName` for

the platform. For example, for an M7i Multiservice Edge Router, the jnxProductNameM7i is .1.3.6.1.4.1.2636.1.1.1.2.10 in the jnxProductName branch, which is identical to the SYSOID of the M7i (.1.3.6.1.4.1.2636.1.1.1.2.10).

How can I determine if a MIB is supported on a platform? How can I determine which MIBs are supported by a device?

MIBs device and platform support is listed on the Junos OS Technical Documentation. See "Standard SNMP MIBs Supported by Junos OS" on page 594 and" Enterprise-Specific SNMP MIBs Supported by Junos OS" on page 622 documents to view the list of MIBs and supported Junos OS devices.

What can I do if the MIB OID query is not responding?

There can be various reasons why the MIB OID query stops responding. One reason could be that the MIB itself is unresponsive. To verify that the MIB responds, use the `show snmp mib walk | get` *`MIB name`* `|` *`MIB OID`* command:

- If the MIB responds, the communication issue exists between the SNMP primary and SNMP agent. Possible reasons for this issue include network issues, an incorrect community configuration, an incorrect SNMP configuration, and so on.
- If the MIB does not respond, enable SNMP `traceoptions` to log PDUs and errors. All incoming and outgoing SNMP PDUs are logged. Check the `traceoptions` output to see if there are any errors.

If you continue to have problems with the MIB OID query, technical product support is available through the Juniper Networks Technical Assistance Center (JTAC).

What is the enterprise branch number for Junos OS?

The enterprise branch number for Junos OS is 2636. Enterprise branch numbers are used in SNMP MIB configurations, and they are also known as SMI network management private enterprise codes.

Which MIB displays the hardware and chassis details on a Juniper Networks device?

The Chassis MIB (jnxchassis.mib) displays the hardware and chassis details for each Juniper Networks device. It provides information about the router and its components. The Chassis MIB objects represent each component and its status.

Which MIB objects can I query to determine the CPU and memory utilization of the Routing Engine, Flexible PIC Concentrator (FPC), and PIC components on a device?

Query the Chassis MIB objects `jnxOperatingMemory`, `jnxOperatingtBuffer`, and `jnxOperatingCPU` to find out the CPU and memory utilization of the hardware components of a device.

Is the interface index (ifIndex) persistent?

The ifIndex is persistent when reboots occur if the Junos OS version remains the same, meaning the values assigned to the interfaces in the ifIndex do not change.

When there is a software upgrade, the device tries to keep the ifIndex persistent on a best effort basis. For Junos OS Release 10.0 and earlier, the ifIndex is not persistent when there is a software upgrade to Junos OS Release 10.1 and later.

Is it possible to set the ifAdminStatus?

SNMP is not allowed to set the ifAdminStatus.

Which MIB objects support SNMP set operations?

The Junos OS SNMP set operations are supported in the following MIB tables and variables:

- snmpCommunityTable
- eventTable
- alarmTable
- snmpTargetAddrExtTable
- jnxPingCtlTable
- pingCtlTable
- traceRouteCtlTable
- jnxTraceRouteCtlTable
- sysContact.0
- sysName.0
- sysLocation.0
- pingMaxConcurrentRequests.0
- traceRouteMaxConcurrentRequests.0
- usmUserSpinLock
- usmUserOwnAuthKeyChange
- usmUserPublic
- vacmSecurityToGroupTable (vacmGroupName, vacmSecurityToGroupStorageType, and vacmSecurityToGroupStatus)
- vacmAccessTable (vacmAccessContextMatch, vacmAccessReadViewName, vacmAccessWriteViewName, vacmAccessNotifyViewName, vacmAccessStorageType, and vacmAccessStatus)

- vacmViewSpinLock
- vacmViewTreeFamilyTable (vacmViewTreeFamilyMask, vacmViewTreeFamilyType, vacmViewTreeFamilyStorageType, and vacmViewTreeFamilyStatus)

Does Junos OS support remote monitoring (RMON)?

Yes, Junos OS supports RMON as defined in RFC 2819, *Remote Network Monitoring Management Information Base*. However, remote monitoring version 2 (RMON 2) is not supported.

Can I use SNMP to determine the health of the processes running on the Routing Engine?

Yes, you can use SNMP to determine the health of the Routing Engine processes by configuring the health monitoring feature. On Juniper Networks devices, RMON alarms and events provide much of the infrastructure needed to reduce the polling overhead from the NMS. However, you must set up the NMS to configure specific MIB objects into RMON alarms. This often requires device-specific expertise and customizing the monitoring application. Additionally, some MIB object instances that need monitoring are set only at initialization, or they 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, such as file system usage, CPU usage, and memory usage, and includes support for unknown or dynamic object instances, such as Junos OS software processes.

To display the health monitoring configuration, use the `show snmp health-monitor` command:

```
user@host> show snmp health-monitor
interval 300;
rising-threshold 90;
falling-threshold 80;
```

When you configure the health monitor, monitoring information for certain object instances is available, as shown in Table 58 on page 663.

Table 58: 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 /.

Table 58: Monitored Object Instances *(Continued)*

Object	Description
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)	Monitor CPU usage for Routing Engines RE0 and RE1. The index values assigned to the Routing Engines depend on whether the Chassis MIB uses a zero-based or a ones-based indexing scheme. Because the indexing scheme is configurable, the correct index is determined whenever the router 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.
jnxOperatingCPU (RE1)	
jnxOperatingBuffer (RE0)	Monitor 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.
jnxOperatingBuffer (RE1)	
sysApplElmtRunCPU	Monitors the CPU usage for each Junos OS software process. Multiple instances of the same process are monitored and indexed separately.
sysApplElmtRunMemory	Monitors the memory usage for each Junos OS software process. Multiple instances of the same process are monitored and indexed separately.

The system log entries generated for any health monitor events, such as thresholds crossed and errors, have a corresponding `HEALTHMONITOR` tag rather than a generic `SNMPD_RMON_EVENTLOG` tag. However, the health monitor sends generic RMON `risingThreshold` and `fallingThreshold` traps.

Are the Ping MIBs returned in decimal notation and ASCII?

Yes, both decimal notation and ASCII are supported, which is the standard implementation in SNMP. All strings are ASCII encoded.

The following example displays the Ping MIB in hexadecimal notation:

```
pingCtlTargetAddress.2.69.72.9.116.99.112.115.97.109.112.108.101 = 0a fa 01 02
```

This translates to ASCII:

```
pingCtlTargetAddress."EH"."tcpsample" = 0a fa 01 02
2= length of the string
69=E
72=H
9=length of second string
116=t
99 =c
112=p
115=s
97=a
109=m
112 =p
108 =l
101 =e
```

As of Junos OS Release 9.6 and later, the Junos OS CLI returns ASCII values using the command `show snmp mib` *`get | get-next | walk`* `ascii`.

The following example shows the output with the ASCII option:

```
user@host> show snmp mib walk pingCtlTargetAddress ascii
pingCtlTargetAddress."EH"."httpgetsample" = http://www.yahoo.com
pingCtlTargetAddress."p1"."t2" = 74 c5 b3 06
pingCtlTargetAddress."p1"."t3" = 74 c5 b2 0c
```

The following example shows the output without the ASCII option:

```
user@host> show snmp mib walk pingCtlTargetAddress
pingCtlTargetAddress.2.69.72.13.104.116.116.112.103.101.116.115.97.109.112.108.101 = http://
www.yahoo.com
pingCtlTargetAddress.2.112.49.2.116.50 = 74 c5 b3 06
pingCtlTargetAddress.2.112.49.2.116.51 = 74 c5 b2 0c
```

You can convert decimal and ASCII values using a decimal ASCII chart like the one at http://www.asciichart.com .

Is IPv6 supported by the Ping MIB for remote operations?

No, IPv6 is not supported.

Is there an SNMP MIB to show Address Resolution Protocol (ARP) table information? Are both IP and MAC addresses displayed in the same table?

Yes, the Junos OS supports the standard MIB `ipNetToMediaTable`, which is described in RFC 2011, *SNMPv2 Management Information Base for the Internet Protocol using SMIv2*. This table is used for mapping IP addresses to their corresponding MAC addresses.

Junos OS SNMP Configuration FAQs

This section presents frequently asked questions and answers related to Junos OS SNMP configuration.

Can the Junos OS be configured for SNMPv1 and SNMPv3 simultaneously?

Yes, SNMP has backward compatibility, meaning that all three versions can be enabled simultaneously.

Can I filter specific SNMP queries on a device?

Yes, you can filter specific SNMP queries on a device using `exclude` and `include` statements.

The following example shows a configuration that blocks read-write operation on all OIDs under .1.3.6.1.2.1.1 for the community `test`:

```
user@host# show snmp
view system-exclude {
    oid .1.3.6.1.2.1.1 exclude;
    oid .1 include;
}
    community test {
        view system-exclude;
        authorization read-write;
    }
```

Can I change the SNMP agent engine ID?

Yes, the SNMP agent engine ID can be changed to the MAC address of the device, the IP address of the device, or any other desired value. Several examples are included here.

The following example shows how to use the MAC address of a device as the SNMP agent engine ID:

```
user@host# show snmp
engine-id {
```

```
    use-mac-address;
}
```

The following example shows how to use the IP address of a device as the SNMP agent engine ID:

```
user@host# show snmp
engine-id {
    use-default-ip-address;
}
```

The following example shows the use of a selected value, AA in this case, as the SNMP agent engine ID of a device:

```
user@host# show snmp
engine-id {
    local AA;
}
```

How can I configure a device with dual Routing Engines or a chassis cluster (SRX Series Services Gateways) for continued communication during a switchover?

When configuring for continued communication, the SNMP configuration should be identical between the Routing Engines. However, it is best to have separate Routing Engine IDs configured for each Routing Engine, especially when using SNMPv3.

The following example shows the configuration of the Routing Engines in a dual Routing Engine device. Notice that the Routing Engine IDs are set to the MAC addresses for each Routing Engine:

```
user@host# show groups
re0 {
    system {
        host-name PE3-re0;
    }
    interfaces {
        fxp0 {
            unit 0 {
                family inet {
                    address 116.197.178.14/27;
                    address 116.197.178.29/27 {
                        master-only;
                    }
```

```
                }
            }
        }
    }
    snmp {
        engine-id {
            use-mac-address;
        }
    }
}
re1 {
    system {
        host-name PE3-re1;
    }
    interfaces {
        fxp0 {
            unit 0 {
                family inet {
                    address 116.197.178.11/27;
                    address 116.197.178.29/27 {
                        master-only;
                    }
                }
            }
        }
    }
    snmp {
        engine-id {
            use-mac-address;
        }
    }
}
```

The following is an example of an SNMPv3 configuration on a dual Routing Engine device:

```
user@host> show snmp name host1
v3 {
    vacm {
        security-to-group {
            security-model usm {
                security-name test123 {
                    group test1;
```

```
                }
                security-name juniper {
                    group test1;
                }
            }
        }
        access {
            group test1 {
                default-context-prefix {
                    security-model any {
                        security-level authentication {
                            read-view all;
                        }
                    }
                }
                context-prefix MGMT_10 {
                    security-model any {
                        security-level authentication {
                            read-view all;
                        }
                    }
                }
            }
        }
    }
    target-address server1 {
        address 116.197.178.20;
        tag-list router1;
        routing-instance MGMT_10;
        target-parameters test;
    }
    target-parameters test {
        parameters {
            message-processing-model v3;
            security-model usm;
            security-level authentication;
            security-name juniper;
        }
        notify-filter filter1;
    }
    notify server {
        type trap;
        tag router1;
```

```
    }
    notify-filter filter1 {
        oid .1 include;
    }
    view all {
        oid .1 include;
    }
    community comm1 {
        view all;
    }
    community comm2;
    community comm3;
    community comm3 {
        view all;
        authorization read-only;
        logical-system LDP-VPLS {
            routing-instance vpls-server1;
        }
    }
    trap-group server1 {
        targets {
            116.197.179.22;
        }
    }
    routing-instance-access;
    traceoptions {
        flag all;
    }
}
```

How can I track SNMP activities?

SNMP trace operations track activity of SNMP agents and record the information in log files.

A sample `traceoptions` configuration might look like this:

```
[edit snmp]
user@host# set traceoptions flag all
```

When the `traceoptions flag all` statement is included at the `[edit snmp]` hierarchy level, the following log files are created:

- snmpd

- mib2d
- rmopd

SNMPv3 FAQs

This section presents frequently asked questions and answers related to SNMPv3.

Why is SNMPv3 important?

SNMP v3 provides enhanced security compared to the other versions of SNMP. It provides authentication and encryption of data. Enhanced security is important for managing devices at remote sites from the management stations.

In my system, the MIB object snmpEngineBoots is not in sync between two Routing Engines in a dual Routing Engine device. Is this normal behavior?

Yes, this is the expected behavior. Each Routing Engine runs its own SNMP process (snmpd), allowing each Routing Engine to maintain its own engine boots. However, if both routing engines have the same engine ID and the routing engine with lesser `snmpEngineBoots` value is selected as the primary routing engine during the switchover process, the `snmpEngineBoots` value of the primary routing engine is synchronized with the `snmpEngineBoots` value of the other routing engine.

Do I need the SNMP manager engine object identifier (OID) for informs?

Yes, the engine OID of the SNMP manager is required for authentication, and informs do not work without it.

I see the configuration of informs under the [edit snmp v3] hierarchy. Does this mean I cannot use informs with SNMPv2c?

Informs can be used with SNMPv2c. The following example shows the basic configuration for SNMPv3 informs on a device (note that the authentication and privacy is set to none):

```
[edit snmp]
v3 {
    usm {
        remote-engine 00000063000100a2c0a845b3 {
            user RU2_v3_sha_none {
                authentication-none;
                privacy-none;
            }
        }
```

```
    }
    vacm {
        security-to-group {
            security-model usm {
                security-name RU2_v3_sha_none {
                    group g1_usm_auth;
                }
            }
        }
        access {
            group g1_usm_auth {
                default-context-prefix {
                    security-model usm {
                        security-level authentication {
                            read-view all;
                            write-view all;
                            notify-view all;
                        }
                    }
                }
            }
        }
    }
    target-address TA2_v3_sha_none {
        address 192.168.69.179;
        tag-list tl1;
        address-mask 255.255.252.0;
        target-parameters TP2_v3_sha_none;
    }
    target-parameters TP2_v3_sha_none {
        parameters {
            message-processing-model v3;
            security-model usm;
            security-level none;
            security-name RU2_v3_sha_none;
        }
        notify-filter nf1;
    }
    notify N1_all_tl1_informs {
        type inform; # Replace “inform” with “trap” to convert informs to traps.
        tag tl1;
    }
    notify-filter nf1 {
```

```
        oid .1 include;
    }
    view all {
        oid .1 include;
    }
}
```

You can convert the SNMPv3 informs to traps by setting the value of the `type` statement at the `[edit snmp v3 notify N1_all_tl1_informs]` hierarchy level to `trap` as shown in the following example:

```
user@host# set snmp v3 notify N1_all_tl1_informs type trap
```

SNMP Interaction with Juniper Networks Devices FAQs

This section presents frequently asked questions and answers related to how SNMP interacts with Juniper Networks devices.

How frequently should a device be polled? What is a good polling rate?

It is difficult to give an absolute number for the rate of SNMP polls per second since the rate depends on the following two factors:

- The number of variable bindings in a protocol data unit (PDU)
- The response time for an interface from the Packet Forwarding Engine

In a normal scenario where no delay is being introduced by the Packet Forwarding Engine and there is one variable per PDU (a Get request), the response time is 130+ responses per second. However, with multiple variables in an SNMP request PDU (30 to 40 for GetBulk requests), the number of responses per second is much less. Because the Packet Forwarding Engine load can vary for each system, there is greater variation in how frequently a device should be polled.

Frequent polling of a large number of counters, especially statistics, can impact the device. We recommend the following optimization on the SNMP managers:

- Use the row-by-row polling method, not the column-by-column method.
- Reduce the number of variable bindings per PDU.
- Increase timeout values in polling and discovery intervals.
- Reduce the incoming packet rate at the SNMP process (snmpd).

For better SNMP response on the device, the Junos OS does the following:

- Filters out duplicate SNMP requests.
- Excludes interfaces that are slow in response from SNMP queries.

One way to determine a rate limit is to note an increase in the `Currently Active` count from the `show snmp statistics extensive` command.

The following is a sample output of the `show snmp statistics extensive` command:

```
user@host> show snmp statistics extensive
  SNMP statistics:
   Input:
     Packets: 226656, Bad versions: 0, Bad community names: 0,
     Bad community uses: 0, ASN parse errors: 0,
     Too bigs: 0, No such names: 0, Bad values: 0,
     Read onlys: 0, General errors: 0,
     Total request varbinds: 1967606, Total set varbinds: 0,
     Get requests: 18478, Get nexts: 75794, Set requests: 0,
     Get responses: 0, Traps: 0,
     Silent drops: 0, Proxy drops: 0, Commit pending drops: 0,
     Throttle drops: 27084, Duplicate request drops: 0
   V3 Input:
     Unknown security models: 0, Invalid messages: 0
     Unknown pdu handlers: 0, Unavailable contexts: 0
     Unknown contexts: 0, Unsupported security levels: 0
     Not in time windows: 0, Unknown user names: 0
     Unknown engine ids: 0, Wrong digests: 0, Decryption errors: 0
   Output:
     Packets: 226537, Too bigs: 0, No such names: 0,
     Bad values: 0, General errors: 0,
     Get requests: 0, Get nexts: 0, Set requests: 0,
     Get responses: 226155, Traps: 382
   SA Control Blocks:
     Total: 222984, Currently Active: 501, Max Active: 501,
     Not found: 0, Timed Out: 0, Max Latency: 25
   SA Registration:
     Registers: 0, Deregisters: 0, Removes: 0
   Trap Queue Stats:
     Current queued: 0, Total queued: 0, Discards: 0, Overflows: 0
   Trap Throttle Stats:
     Current throttled: 0, Throttles needed: 0
```

```
Snmp Set Stats:
  Commit pending failures: 0, Config lock failures: 0
  Rpc failures: 0, Journal write failures: 0
  Mgd connect failures: 0, General commit failures: 0
```

Does SNMP open dynamic UDP ports? Why?

The SNMP process opens two additional ports (sockets): one for IPv4 and one for IPv6. This enables the SNMP process to send traps.

I am unable to perform a MIB walk on the ifIndex. Why is this?

Any variable bindings or values with an access level of `not-accessible` cannot be queried directly because they are part of other variable bindings in the SNMP MIB table. The ifIndex has an access level of `not-accessible`. Therefore, it cannot be accessed directly because it is part of the variable bindings. However, the ifIndex can be accessed indirectly through the variable bindings.

I see SNMP_IPC_READ_ERROR messages when the SNMP process restarts on my system and also during Routing Engine switchover. Is this acceptable?

Yes, it is acceptable to see `SNMP_IPC_READ_ERROR` messages when the SNMP process is restarted, the system reboots, or during a Routing Engine switchover. If all the processes come up successfully and the SNMP operations are working properly, then these messages can be ignored.

What is the source IP address used in the response PDUs for SNMP requests? Can this be configured?

The source IP address used in the response PDUs for SNMP requests is the IP address of the outgoing interface to reach the destination. The source IP address cannot be configured for responses. It can only be configured for traps.

SNMP Traps and Informs FAQs

This section presents frequently asked questions and answers related to SNMP traps and informs.

Does the Junos OS impose any rate limiting on SNMP trap generation?

The Junos OS implements a trap-queuing mechanism to limit the number of traps that are generated and sent.

If a 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, 2, 4, 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 traps in the queue are deleted.

Junos OS also has a throttle threshold mechanism to control the number of traps sent (default 500 traps) during a particular throttle interval (default 5 seconds). This helps ensure consistency in trap traffic, especially when a large number of traps are generated due to interface status changes.

The throttle interval begins when the first trap arrives at the throttle. All traps within the throttle threshold value are processed, and traps exceeding the threshold value are queued. The maximum size of all trap queues (the throttle queue and the destination queue) is 40,000 traps. The maximum size of any one queue is 20,000 traps. When a trap is added to the throttle queue, or if the throttle queue has exceeded the maximum size, the trap is moved to the top of the destination queue. Further attempts to send the trap from the destination queue are stopped for a 30-second period, after which the destination queue restarts sending the traps.

NOTE: For the Juniper Networks EX Series Ethernet Switch, the maximum size of all trap queues (the throttle queue and the destination queue) is 1,000 traps. The maximum size for any one queue on the EX Series is 500 traps.

I did not see a trap when I had a syslog entry with a critical severity. Is this normal? Can it be changed?

Not every syslog entry with critical severity is a trap. However, you can convert any syslog entry to a trap using the `event-options` statement.

The following example shows how to configure a `jnxSyslogTrap` whenever an `rpd_ldp_nbrdown` syslog entry message error occurs.

```
user@host> show event-options
policy snmptrap {
    events rpd_ldp_nbrdown;
    then {
        raise-trap;
    }
}
```

Are SNMP traps compliant with the Alarm Reporting Function (X.733) on the Junos OS?

No, SNMP traps on the Junos OS are not X.733 compliant.

Can I set up filters for traps or informs?

Traps and informs can be filtered based on the trap category and the object identifier. You can specify categories of traps to receive per host by using the `categories` statement at the [`edit snmp trap-group` *trap-group*] hierarchy level. 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;
    }
}
```

The Junos OS also has a more advanced filter option (`notify-filter`) for filtering specific traps or a group of traps based on their object identifiers.

The SNMPv3 configuration also supports filtering of SNMPv1 and SNMPv2 traps and excluding Juniper Networks enterprise-specific configuration management traps, as shown in the following configuration 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;
}
}
```

Can I simulate traps on a device?

Yes, you can use the `request snmp spoof-trap` *`trap name`* command for simulating a trap to the NMS that normally receives your device's traps. You can also add required values using the `variable-bindings` parameter.

The following example shows how to simulate a trap to the local NMS using variable bindings:

```
user@host> request snmp spoof-trap linkDown variable-bindings "ifIndex[116]=116,
ifAdminStatus[116]=1 ,ifOperStatus[116]=2 , ifName[116]=ge-1/0/1"
```

How do I generate a warm start SNMPv1 trap?

When the SNMP process is restarted under normal conditions, a warm start trap is generated if the system up time is more than 5 minutes. If the system up time is less than 5 minutes, a cold start trap is generated.

The NMS sees only the MIB OIDs and numbers, but not the names of the SNMP traps. Why?

Before the NMS can recognize the SNMP trap details, such as the names of the traps, it must first compile and understand the MIBs and then parse the MIB OIDs.

In the Junos OS, how can I determine to which category a trap belongs?

For a list of common traps and their categories, see SNMP MIB Explorer .

Can I configure a trap to include the source IP address?

Yes, you can configure the `source-address`, `routing-instance`, or `logical-instance` name for the source IP address using the `trap-options` command:

```
user@host> show snmp trap-options
source-address 10.1.1.1;
```

Can I create a custom trap?

Yes, you can use the `jnxEventTrap` event script to create customized traps as needed.

In the following example, a Junos OS operations (op) script is triggered when a `UI_COMMIT_NOT_CONFIRMED` event is received. The Junos OS op script matches the complete message of the event and generates an SNMP trap.

Example: Junos OS Op Script

```
version 1.0;

ns junos = "http://xml.juniper.net/junos/*/junos";
ns xnm = "http://xml.juniper.net/xnm/1.1/xnm";
ns jcs = "http://xml.juniper.net/junos/commit-scripts/1.0";
```

```
param $event;
param $message;

match / {

    /*
     * trapm utilty wants the following characters in the value to be escaped
     * '[', ']', ' ', '=', and ','
     */
    var $event-escaped = {
        call escape-string($text = $event, $vec = '[] =,');
    }

    var $message-escaped = {
        call escape-string($text = $message, $vec = '[] =,');
    }

    <op-script-results> {
    var $rpc = <request-snmp-spoof-trap> {
        <trap> "jnxEventTrap";
        <variable-bindings> "jnxEventTrapDescr[0]='Event-Trap' , "
        _ "jnxEventAvAttribute[1]='event' , "
        _ "jnxEventAvValue[1]='" _ $event-escaped _ "' , "
        _ "jnxEventAvAttribute[2]='message' , "
        _ "jnxEventAvValue[1]='" _ $message-escaped _ "'";
    }

    var $res = jcs:invoke($rpc);
    }
}

template escape-string ($text, $vec) {

    if (jcs:empty($vec)) {
        expr $text;

    } else {
        var $index = 1;
        var $from = substring($vec, $index, 1);
        var $changed-value = {
            call replace-string($text, $from) {
                with $to = {
                    expr "\\";
```

```
                    expr $from;
                }
            }
        }

        call escape-string($text = $changed-value, $vec = substring($vec, $index
 + 1));
    }
}

template replace-string ($text, $from, $to) {

    if (contains($text, $from)) {
        var $before = substring-before($text, $from);
        var $after = substring-after($text, $from);
        var $prefix = $before _  $to;

        expr $before;
        expr $to;
        call replace-string($text = $after, $from, $to);

    } else {
        expr $text;
    }
}
```

After creating your customized trap, you must configure a policy on your device to tell the device what actions to take after it receives the trap.

Here is an example of a configured policy under the `[edit event-options]` hierarchy:

```
[edit event-options]
user@host> show
policy trap-on-event {
    events UI_COMMIT_NOT_CONFIRMED;
    attributes-match {
        UI_COMMIT_NOT_CONFIRMED.message matches complete;
    }
    then {
        event-script ev-syslog-trap.junos-op {
            arguments {
```

```
                event UI_COMMIT_NOT_CONFIRMED;
                message "{$$.message}";
            }
        }
    }
}
```

Can I disable link up and link down traps on interfaces?

Yes, link up and link down traps can be disabled in the interface configuration. To disable the traps, use the `no-traps` statement at the [`edit interfaces` *`interface-name`* `unit` *`logical-unit-number`*] and [`edit logical-systems` *`logical-system-name`* `interfaces` *`interface-name`* `unit` *`logical-unit-number`*] hierarchies for physical and logical interfaces.

```
(traps | no-traps);
```

I see the link up traps on logical interfaces, but I do not see the link down traps. Is this normal behavior?

For Ethernet and ATM types of interfaces, Junos OS does not send link down traps for a logical interface if the physical interface is down to prevent flooding alarms for the same root cause. However, when the physical interface and logical interfaces come back up, traps are sent indicating link up. This is because the physical interface coming up does not necessarily mean the logical interfaces are also coming up.

For SONET types of interfaces with PPP encapsulation, Junos OS does send link down traps for a logical interface if the physical interface is down. When the physical interface and logical interfaces come back up, traps are sent for both the physical and logical interfaces indicating link up.

For SONET types of interfaces with HDLC encapsulation, Junos OS does not send link down traps for a logical interface if the physical interface is down. When the physical interface and logical interfaces come back up, traps are sent for both the physical and logical interfaces indicating link up.

For channelize interfaces with PPP encapsulation, Junos OS does send link down traps for a logical interface if the physical interface is down. When the physical interface and logical interfaces come back up, traps are sent for both the physical and logical interfaces indicating link up.

For channelize interfaces with HDLC encapsulation, Junos OS does not send link down traps for a logical interface if the physical interface is down. When the physical interface and logical interfaces come back up, traps are sent for both the physical and logical interfaces indicating link up.

Junos OS Dual Routing Engine Configuration FAQs

This section presents frequently asked questions and answers related to the configuration of dual Routing Engines.

The SNMP configuration should be identical between the Routing Engines when configuring for continued communication. However, we recommend having separate Routing Engine IDs configured for each Routing Engine, when using SNMPv3.

In my system, the MIB object snmpEngineBoots is not in sync between two Routing Engines in a dual Routing Engine device. Is this normal behavior?

Yes. This is the normal behavior. Each Routing Engine runs its own SNMP process (snmpd) agent, allowing each Routing Engine to maintain its own engine boots.

Is there a way to identify that an address belongs to RE0, RE1, or the master Routing Engine management interface (fxp0) by looking at an SNMP walk?

No. When you do an SNMP walk on the device, it only displays the primary Routing Engine management interface address.

What is the best way to tell if the current IP address belongs to fxp0 or a Routing Engine, from a CLI session?

Routing Engines are mapped with the `fxp0` interface. This means that when you query RE0, the ifTable reports the `fxp0` interface address of RE0 only. Similarly, if you query RE1, the ifTable reports the `fxp0` interface address of RE1 only.

When there is a failover, the master hostname is changed since the hostname belongs to the Routing Engine. Is this correct?

Yes. You can configure the same hostname or different hostnames. Either would work.

If only the primary IP address is configured (for example, 192.168.2.5), and the `sysDescr.0` object has the same string configured on both of the Routing Engines, then even after a switchover, the `sysDescr.0` object returns the same value. The following sample shows the results you get by using the `snmpget` command:

```
bng-junos-pool02: /c/svivek/PR_BRANCH/src> snmpget -c jnpr -v2c 192.168.2.5
sysDescr.0 system.sysDescr.0 = foo
```

SNMP Support for Routing Instances FAQs

This section presents frequently asked questions and answers related to how SNMP supports routing instances.

Can the SNMP manager access data for routing instances?

Yes, the 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.

Two different routing instance behaviors can occur, depending on where the clients originate:

- 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.

Routing instances are identified by either the context field in SNMPv3 requests or encoded in the community string in SNMPv1 or SNMPv2c 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 (including views, source address restrictions, and access privileges) 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 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.

Additionally, 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 SNMPv3 requests, the name *`logical system/routing instance`* should be identified directly in the context field.

Can I access a list of all routing instances on a device?

Yes, you can access a list of all the routing instances on a device using the vacmContextName object in the SNMP-VIEW-BASED-ACM MIB. In SNMP, each routing instance becomes a VACM context; this is why the routing instances appear in the vacmContextName object.

Can I access a default routing instance from a client in another logical router or routing instance?

No, the SNMP agent can only access data of the logical router to which it is connected.

SNMP Counters FAQs

This section presents frequently asked questions and answers related to SNMP counters.

Which MIB should I use for interface counters?

Interface management over SNMP is based on two tables: the `ifTable` and its extension the `ifXTable`. Both are described in RFC 1213, *Management Information Base for Network Management of TCP/IP-based internets: MIB-II* and RFC 2233, *The Interfaces Group MIB using SMIv2*.

Interfaces can have several layers, depending on the media, and each sublayer is represented by a separate row in the table. The relationship between the higher layer and lower layers is described in the `ifStackTable`.

The `ifTable` defines 32-bit counters for inbound and outbound octets (ifInOctets/ifOutOctets), packets (ifInUcastPkts/ifOutUcastPkts, ifInNUcastPkts /ifOutNUcastPkts), errors, and discards.

The `ifXTable` provides similar 64-bit counters, also called high capacity (HC) counters, for inbound and outbound octets (ifHCInOctets/ifHCOutOctets) and inbound packets (ifHCInUcastPkts).

When should 64-bit counters be used?

It is always good to use 64-bit counters because they contain statistics for both low and high capacity components.

Are the SNMP counters ifInOctets and ifOutOctets the same as the command reference show interfaces statistics in and out counters?

Yes, these are the same, but only if SNMP is enabled when the router boots up. If you power on a Juniper Networks device and then enable SNMP, the SNMP counters start from 0. SNMP counters do not automatically receive their statistics from the `show` command output. Similarly, using the `clear statistics` command does not clear the statistics that the SNMP counters collected, which can cause a discrepancy in the data that is seen by both processes.

Do the SNMP counters ifInOctets and ifOutOctets include the framing overhead for Point-to-Point Protocol (PPP) and High-Level Data Link Control (HDLC)?

Yes.

4

PART

Remote Network Monitoring (RMON) with SNMP Alarms and Events

Remote Network Monitoring (RMON) | 687

Configure RMON History Sampling | 715

Monitor Network Service Quality by using RMON | 717

Health Monitoring with SNMP | 750

Remote Network Monitoring (RMON)

SUMMARY

This section describes how Junos OS supports the *Remote Network Monitoring* (RMON) MIB (RFC 2819) that allows a management device to monitor the values of MIB objects, or variables, against configured thresholds. When the value of a variable crosses a threshold, an alarm and its corresponding event are generated. The event can be logged and can generate an SNMP trap.

IN THIS SECTION

- RMON Overview | **687**
- RMON Alarms and Events Configuration | **692**
- Configure RMON Alarms and Events | **692**
- Monitor RMON MIB Tables | **696**
- RMON MIB Event, Alarm, Log, and History Control Tables | **697**
- Minimum RMON Alarm and Event Entry Configuration | **700**
- Configure an RMON Alarm Entry and Its Attributes | **701**
- Configure an RMON Event Entry and Its Attributes | **706**
- Example: Configure an RMON Alarm and Event Entry | **707**
- Use alarmTable to Monitor MIB Objects | **707**
- Use eventTable to Log Alarms | **712**

RMON Overview

IN THIS SECTION

- RMON Alarms | **688**
- RMON Events | **690**
- Alarm Thresholds and Events | **691**

An operational support system (OSS) or a fault-monitoring system can be used to automatically monitor events that track many different metrics, including performance, availability, faults, and environmental data. For example, an administrator might want to know when the internal temperature of a chassis has risen above a configured threshold, which might indicate that a chassis fan tray is faulty, the chassis air flow is impeded, or the facility cooling system in the vicinity of the chassis is not operating normally.

The RMON MIB also defines tables that store various statistics for Ethernet interfaces, including the `etherStatsTable` and the `etherHistoryTable`. The `etherStatsTable` contains cumulative real-time statistics for Ethernet interfaces, such as the number of unicast, multicast, and broadcast packets received on an interface. The `etherHistoryTable` maintains a historical sample of statistics for Ethernet interfaces. The control of the `etherHistoryTable`, including the interfaces to track and the sampling interval, is defined by the RMON `historyControlTable`.

To enable RMON alarms, you perform the following steps:

1. Configure SNMP, including trap groups. You configure SNMP at the [`edit snmp`] hierarchy level.
2. Configure rising and falling events in the `eventTable`, including the event types and trap groups. You can also configure events using the CLI at the [`edit snmp rmon event`] hierarchy level.
3. Configure alarms in the `alarmTable`, including the variables to monitor, rising and falling thresholds, the sampling types and intervals, and the corresponding events to generate when alarms occur. You can also configure alarms using the CLI at the [`edit snmp rmon alarm`] hierarchy level.

   Extensions to the `alarmTable` are defined in the Juniper Networks enterprise-specific MIB jnxRmon (mib-jnx-rmon.txt).

This topic covers the following sections:

RMON Alarms

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 https://www.juniper.net/documentation/en_US/junos16.1/topics/reference/mibs/mib-jnx-rmon.txt.

RMON Events

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

eventTable contains the following objects:

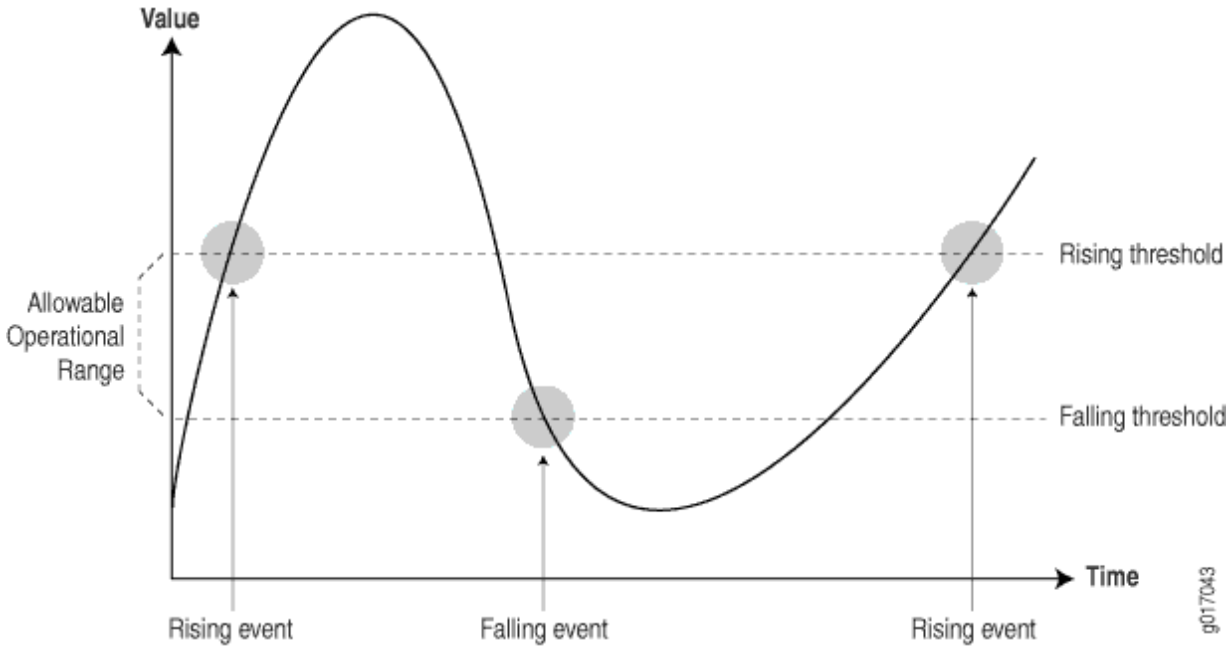
- eventIndex—An index that uniquely identifies an entry in eventTable. Each entry defines one event that is generated when the appropriate conditions occur.
- eventDescription—A comment describing the event entry.
- eventType—Type of notification that the probe makes about this event.
- eventCommunity—Trap 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.

Alarm Thresholds and Events

By setting a rising and a falling threshold for a monitored variable, you can be alerted whenever the value of the variable falls outside the allowable operational range (see Figure 24 on page 691).

Figure 24: Setting Thresholds



Events are only generated when the alarm threshold is first crossed in any one direction rather than after each sample interval. For example, if a rising threshold alarm, along with its corresponding event, is raised, no more threshold crossing events occur until a corresponding falling alarm occurs. This considerably reduces the quantity of events that are produced by the system, making it easier for operations staff to react when events do occur.

Before you 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 3 months is not unusual when you first identify the operational ranges and define thresholds, but baseline monitoring should continue over the life span of each monitored variable.

SEE ALSO

- Juniper Networks Enterprise-Specific MIBs

RMON Alarms and Events Configuration

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;
    }
}
```

Configure RMON Alarms and Events

IN THIS SECTION

- Configure SNMP | 693

Configure an Event | 694
Configure an Alarm | 695

The Junos OS supports the *Remote Network Monitoring* (RMON) MIB (RFC 2819). This allows a management device to monitor the values of MIB objects, or variables, against configured thresholds. When the value of a variable crosses a threshold, an alarm and its corresponding event are generated. The event can be logged and can generate an SNMP trap.

To configure RMON alarms and events using the CLI, perform these tasks:

Configure SNMP

To configure SNMP:

1. Grant read-only access to all SNMP clients:

```
[edit snmp]
user@switch# set community community-name authorization authorization
```

For example:

```
[edit snmp]
user@switch# set community public authorization read-only
```

2. Grant read-write access to the RMON and jnx-rmon MIBs:

```
[edit snmp]
user@switch# set view view-name oid object-identifier include
user@switch# set view view-name oid object-identifier include
user@switch# set community community-name authorization authorization view view-name
```

For example:

```
[edit snmp]
user@switch# set view rmon-mib-view oid .1.3.6.1.2.1.16 include
user@switch# set view rmon-mib-view oid .1.3.6.1.4.1.2636.13 include
user@switch# set community private authorization read-write view rmon-mib-view
```

OIDs 1.3.6.1.2.1.16 and 1.3.6.1.4.1.2636.13 correspond to the RMON and jnxRmon MIBs.

3. Configure an SNMP trap group:

```
[edit snmp]
user@switch# set trap-group group-name categories category
user@switch# set trap-group group-name targets address
```

For example:

```
[edit snmp]
user@switch# set trap-group rmon-trap-group categories rmon-alarm
user@switch# set trap-group rmon-trap-group targets 192.168.5.5
```

The trap group rmon-trap-group is configured to send RMON traps to 192.168.5.5.

Configure an Event

To configure an event:

1. Configure an event index, community name, and type:

```
[edit snmp rmon]
user@switch# set event index community community-name  typetype
```

For example:

```
[edit snmp rmon]
user@switch# set event 1 community rmon-trap-group type log-and-trap
```

The event community corresponds to the SNMP trap group and is not the same as an SNMP community. This event generates an SNMP trap and adds an entry to the logTable in the RMON MIB.

2. Configure a description for the event:

```
[edit snmp rmon]
user@switch# set event index description description
```

For example:

```
[edit snmp rmon]
user@switch# set event 1 description “rmon event”
```

Configure an Alarm

To configure an alarm:

1. Configure an alarm index, the variable to monitor, the rising and falling thresholds, and the corresponding rising and falling events:

   ```
   [edit snmp rmon]
   user@switch# set alarm index variable oid-variable falling-threshold integer  rising-
   threshold integer rising-event-index index falling-event-index index
   ```

   For example:

   ```
   [edit snmp rmon]
   user@switch# set alarm 5 variable .1.3.6.1.4.1.2636.3.1.13.1.8.9.1.0.0 falling-threshold 75
   rising-threshold 90 rising-event-index 1 falling-event-index 1
   ```

   The variable .1.3.6.1.4.1.2636.3.1.13.1.8.9.1.0.0 corresponds to the jnxRmon MIB object jnxOperatingCPU, which represents the CPU utilization of the Routing Engine. The falling and rising threshold integers are 75 and 90. The rising and falling events both generate the same event (event index 1).

2. Configure the sample interval and type and the alarm type:

   ```
   [edit snmp rmon]
   user@switch# set alarm index  interval seconds sample-type (absolute-value | delta-value)
   startup-alarm (falling-alarm | rising-alarm | rising-or-falling-alarm)
   ```

   For example:

   ```
   [edit snmp rmon]
   user@switch# set alarm 5 interval 30 sample-type absolute-value startup-alarm rising-or-
   falling-alarm
   ```

The absolute value of the monitored variable is sampled every 30 seconds. The initial alarm can occur because of rising above the rising threshold or falling below the falling threshold.

Monitor RMON MIB Tables

IN THIS SECTION

- Purpose | 696
- Action | 696
- Meaning | 697

Purpose

Monitor remote monitoring (RMON) alarm, event, and log tables.

Action

To display the RMON tables:

```
user@switch> show snmp rmon
Alarm
Index  Variable description                          Value State

     5 monitor
       jnxOperatingCPU.9.1.0.0                           5 falling threshold

Event
Index  Type                        Last Event
    1  log and trap                2010-07-10 11:34:17 PDT
Event Index: 1
    Description: Event 1 triggered by Alarm 5, rising threshold (90) crossed, (variable:
jnxOperatingCPU.9.1.0.0, value: 100)
    Time: 2010-07-10 11:34:07 PDT
    Description: Event 1 triggered by Alarm 5, falling threshold (75) crossed, (variable:
```

```
jnxOperatingCPU.9.1.0.0, value: 5)
    Time: 2010-07-10 11:34:17 PDT
```

Meaning

The display shows that an alarm has been defined to monitor jnxRmon MIB object jnxOperatingCPU, which represents the CPU utilization of the Routing Engine. The alarm is configured to generate an event that sends an SNMP trap and adds an entry to the logTable in the RMON MIB. The log table shows that two occurrences of the event have been generated—one for rising above a threshold of 90 percent, and one for falling below a threshold of 75 percent.

SEE ALSO

- Configuring RMON Alarms and Events
- show snmp rmon
- show snmp rmon history
- clear snmp statistics
- clear snmp history

RMON MIB Event, Alarm, Log, and History Control Tables

Table 59 on page 697 provides each field in the RMON eventTable, the description of the field, and the corresponding Junos OS statement that you can use to configure the field. The Junos OS statements reside at the [`edit snmp rmon`] hierarchy level.

Table 59: RMON Event Table

Field	Description	Statement [edit snmp rmon]
eventDescription	Text description of this event.	`description`
eventType	Type of event (for example, log, trap, or log and trap).	`type`

Table 59: RMON Event Table *(Continued)*

Field	Description	Statement [edit snmp rmon]
eventCommunity	Trap group to which to send this event, as defined in the Junos OS configuration. (This is not the same as the SNMP community.)	`community`
eventOwner	Entity (for example, manager) that created this event.	—
eventStatus	Status of this row (for example, valid, invalid, or createRequest).	—

Table 60 on page 698 provides each field in the RMON alarmTable, the description of the field, and the corresponding Junos OS statement that you can use to configure the field. The Junos OS statements reside at the [`edit snmp rmon`] hierarchy level.

Table 60: RMON Alarm Table

Field	Description	Statement [edit snmp rmon]
alarmStatus	Status of this row (for example, valid, invalid, or createRequest)	—
alarmInterval	Sampling period (in seconds) of the monitored variable	`interval`
alarmVariable	Object identifier (OID) and instance of the variable to be monitored	—
alarmValue	Actual value of the sampled variable	—
alarmSampleType	Sample type (absolute or delta changes)	`sample-type`
alarmStartupAlarm	Initial alarm (rising, falling, or either)	`startup-alarm`

Table 60: RMON Alarm Table *(Continued)*

Field	Description	Statement [edit snmp rmon]
alarmRisingThreshold	Rising threshold against which to compare the value	`rising-threshold`
alarmFallingThreshold	Falling threshold against which to compare the value	`falling-threshold`
alarmRisingEventIndex	Index (row) of the rising event in the event table	`rising-event-index`
alarmFallingEventIndex	Index (row) of the falling event in the event table	`falling-event-index`

Table 61 on page 699 provides each field in the jnxRmon jnxRmonAlarmTable, which is an extension to the RMON alarmTable. You can troubleshoot the RMON agent, rmopd, that runs on a switch by inspecting the contents of the jnxRmonAlarmTable object.

Table 61: jnxRmon Alarm Table

Field	Description
jnxRmonAlarmGetFailCnt	Number of times the internal `Get` request for the variable failed
jnxRmonAlarmGetFailTime	Value of the sysUpTime object when the last failure occurred
jnxRmonAlarmGetFailReason	Reason why the `Get` request failed
jnxRmonAlarmGetOkTime	Value of the sysUpTime object when the variable moved out of failure state
jnxRmonAlarmState	Status of this alarm entry

Table 62 on page 700 provides each field in the RMON historyControlTable, the description of the field, and the corresponding Junos OS statement that you can use to configure the field. The Junos OS statements reside at the [`edit snmp rmon history`] hierarchy level. The historyControlTable controls the RMON etherHistoryTable.

Table 62: RMON History Control Table

Field	Description	Statement [edit snmp rmon history]
historyControlDataSource	Identifies the source of the data for which historical data was collected.	`interface`
historyControlBucketsRequested	Requested number of discrete time intervals over which data is to be saved.	`bucket-size`
historyControlBucketsGranted	Number of discrete sampling intervals over which data is to be saved.	—
historyControlInterval	Interval, in seconds, over which the data is sampled for each bucket.	`interval`
historyControlOwner	Entity that configured this entry.	`owner`
historyControlStatus	Status of this entry.	—

Minimum RMON Alarm and Event Entry Configuration

To enable RMON on the router, you must configure an alarm entry and an event entry. To do this, include the following statements at the `[edit snmp rmon]` hierarchy level:

```
[edit snmp rmon]
alarm index {
    rising-event-index index;
    rising-threshold integer;
    sample-type type;
    variable oid-variable;
}
event index;
```

Configure an RMON Alarm Entry and Its Attributes

IN THIS SECTION

- Configure the Alarm Entry | 701
- Configure the Description | 702
- Configure the Falling Event Index or Rising Event Index | 702
- Configure the Falling Threshold or Rising Threshold | 702
- Configure the Interval | 703
- Configure the Falling Threshold Interval | 703
- Configure the Request Type | 704
- Configure the Sample Type | 704
- Configure the Startup Alarm | 705
- Configure the System Log Tag | 705
- Configure the Variable | 705

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:

Configure 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.

Configure 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;
```

Configure 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.

Configure 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.

Configure 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.

Configure 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.

Configure 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.

Configure 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.

Configure 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`.

Configure 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;
```

Configure 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).

Configure an RMON Event Entry and Its Attributes

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.

Example: Configure an RMON Alarm and Event Entry

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;
    }
}
```

Use alarmTable to Monitor MIB Objects

IN THIS SECTION

- Create an Alarm Entry | **708**
- Configure the Alarm MIB Objects | **708**
- Activate a New Row in alarmTable | **711**
- Modify an Active Row in alarmTable | **711**
- Deactivate a Row in alarmTable | **712**

To use alarmTable to monitor a MIB object, perform the following tasks:

Create an Alarm Entry

To create an alarm entry, first create a new row in alarmTable using the alarmStatus object. For example, create alarm #1 using the UCD command-line utilities:

```
snmpset -Os -v2c router community alarmStatus.1 i createRequest
```

Configure the Alarm MIB Objects

IN THIS SECTION

- alarmInterval | **709**
- alarmVariable | **709**
- alarmSampleType | **709**
- alarmValue | **709**
- alarmStartupAlarm | **710**
- alarmRisingThreshold | **710**
- alarmFallingThreshold | **710**
- alarmOwner | **711**
- alarmRisingEventIndex | **711**
- alarmFallingEventIndex | **711**

Once you have created the new row in alarmTable, configure the following Alarm MIB objects:

NOTE: Other than alarmStatus, you cannot modify any of the objects in the entry if the associated alarmStatus object is set to `valid`.

alarmInterval

The interval, in seconds, over which data is sampled and compared with the rising and falling thresholds. For example, to set `alarmInterval` for alarm #1 to 30 seconds, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmInterval.1 i 30
```

alarmVariable

The object identifier of the variable to be sampled. During a `Set` request, if the supplied variable name is not available in the selected MIB view, a badValue error is returned. If at any time the variable name of an established alarmEntry is no longer available in the selected MIB view, the probe changes the status of alarmVariable to invalid. For example, to identify ifInOctets.61 as the variable to be monitored, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmVariable.1 o .1.3.6.1.2.1.2.2.1.10.61
```

alarmSampleType

The method of sampling the selected variable and calculating the value to be compared against the thresholds. If the value of this object is absoluteValue, the value of the selected variable is compared directly with the thresholds at the end of the sampling interval. If the value of this object is deltaValue, the value of the selected variable at the last sample is subtracted from the current value, and the difference is compared with the thresholds. For example, to set alarmSampleType for alarm #1 to deltaValue, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmSampleType.1 i deltaValue
```

alarmValue

The value of the variable during the last sampling period. This value is compared with the rising and falling thresholds. If the sample type is deltaValue, this value equals the difference between the samples at the beginning and end of the period. If the sample type is `absoluteValue`, this value equals the sampled value at the end of the period.

alarmStartupAlarm

An alarm that is sent when this entry is first set to valid. If the first sample after this entry becomes valid is greater than or equal to `risingThreshold`, and `alarmStartupAlarm` is equal to `risingAlarm` or `risingOrFallingAlarm`, then a single rising alarm is generated. If the first sample after this entry becomes valid is less than or equal to `fallingThreshold` and `alarmStartupAlarm` is equal to `fallingAlarm` or `risingOrFallingAlarm`, then a single falling alarm is generated. For example, to set `alarmStartupAlarm` for alarm #1 to `risingOrFallingAlarm`, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmStartupAlarm.1 i risingOrFallingAlarm
```

alarmRisingThreshold

A 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 `alarmStartupAlarm` is equal to `risingAlarm` or `risingOrFallingAlarm`. After a rising event is generated, another rising event cannot be generated until the sampled value falls below this threshold and reaches `alarmFallingThreshold`. For example, to set `alarmRisingThreshold` for alarm #1 to `100000`, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmRisingThreshold.1 i 100000
```

alarmFallingThreshold

A 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 `alarmStartupAlarm` is equal to `fallingAlarm` or `risingOrFallingAlarm`. After a falling event is generated, another falling event cannot be generated until the sampled value rises above this threshold and reaches `alarmRisingThreshold`. For example, to set `alarmFallingThreshold` for alarm #1 to `10000`, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmFallingThreshold.1 i 10000
```

alarmOwner

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.

alarmRisingEventIndex

The index of the eventEntry object that is used when a rising threshold is crossed. If there is no corresponding entry in eventTable, then no association exists. If this value is zero, no associated event is generated because zero is not a valid event index. For example, to set alarmRisingEventIndex for alarm #1 to `10`, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmRisingEventIndex.1 i 10
```

alarmFallingEventIndex

The index of the eventEntry object that is used when a falling threshold is crossed. If there is no corresponding entry in eventTable, then no association exists. If this value is zero, no associated event is generated because zero is not a valid event index. For example, to set alarmFallingEventIndex for alarm #1 to `10`, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community alarmFallingEventIndex.1 i 10
```

Activate a New Row in alarmTable

To activate a new row in alarmTable, set alarmStatus to `valid` using an SNMP `Set` request:

```
snmpset -Os -v2c router community alarmStatus.1 i valid
```

Modify an Active Row in alarmTable

To modify an active row, first set alarmStatus to underCreation using an SNMP `Set` request:

```
snmpset -Os -v2c router community alarmStatus.1 i underCreation
```

Then change the row contents using an SNMP `Set` request:

```
snmpset -Os -v2c router community alarmFallingThreshold.1 i 1000
```

Finally, activate the row by setting alarmStatus to `valid` using an SNMP `Set` request:

```
snmpset -Os -v2c router community alarmStatus.1 i valid
```

Deactivate a Row in alarmTable

To deactivate a row in alarmTable, set alarmStatus to `invalid` using an SNMP `Set` request:

```
snmpset -Os -v2c router community alarmStatus.1 i invalid
```

Use eventTable to Log Alarms

IN THIS SECTION

- Create an Event Entry | **712**
- Configure the MIB Objects | **713**
- Activate a New Row in eventTable | **715**
- Deactivate a Row in eventTable | **715**

To use eventTable to log alarms, perform the following tasks:

Create an Event Entry

The RMON eventTable controls the generation of notifications from the router. Notifications can be logs (entries to logTable and syslogs) or SNMP traps. Each event entry can be configured to generate any combination of these notifications (or no notification). When an event specifies that an SNMP trap is to be generated, the trap group that is used when sending the trap is specified by the value of the associated eventCommunity object. Consequently, the community in the trap message will match the value specified by eventCommunity. If nothing is configured for eventCommunity, a trap is sent using each trap group that has the rmon-alarm category configured.

Configure the MIB Objects

IN THIS SECTION

- eventType | 713
- eventCommunity | 713
- eventOwner | 714
- eventDescription | 714

Once you have created the new row in eventTable, set the following objects:

NOTE: The eventType object is required. All other objects are optional.

eventType

The type of notification that the router generates when the event is triggered.

This object can be set to the following values:

- `log`—Adds the event entry to logTable.
- `log-and-trap`—Sends an SNMP trap and creates a log entry.
- `none`—Sends no notification.
- `snmptrap`—Sends an SNMP trap.

For example, to set `eventType` for event #1 to `log-and-trap`, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community eventType.1 i log-and-trap
```

eventCommunity

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. For example, to set eventCommunity for event #1 to boy-elroy, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community eventCommunity.1 s "boy-elroy"
```

NOTE: The eventCommunity object is optional. If you do not set this object, then the field is left blank.

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.

For example, to set eventOwner for event #1 to george jetson, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community eventOwner.1 s "george jetson"
```

NOTE: The eventOwner object is optional. If you do not set this object, then the field is left blank.

eventDescription

Any text string specified by the creating management application or the command-line interface (CLI). The use of this string is application dependent.

For example, to set eventDescription for event #1 to spacelys sprockets, use the following SNMP `Set` request:

```
snmpset -Os -v2c router community eventDescription.1 s "spacelys sprockets"
```

NOTE: The eventDescription object is optional. If you do not set this object, then the field is left blank.

Activate a New Row in eventTable

To activate the new row in eventTable, set eventStatus to `valid` using an SNMP `Set` request such as:

```
snmpset -Os -v2c router community eventStatus.1 i valid
```

Deactivate a Row in eventTable

To deactivate a row in eventTable, set eventStatus to `invalid` using an SNMP `Set` request such as:

```
snmpset -Os -v2c router community eventStatus.1 i invalid
```

Configure RMON History Sampling

IN THIS SECTION

- Configure RMON History Sampling Collection | **715**
- View and Clear RMON History Statistics | **716**

The Junos OS supports the history control group (`etherHistoryTable`) of the *Remote Network Monitoring* (RMON) MIB (RFC 2819). The history control tables record statistical samples from an Ethernet network and store them for later retrieval.

To configure RMON history sampling and view or clear collected statistics using the Junos OS CLI, perform the following tasks:

Configure RMON History Sampling Collection

Use the `history` statement at the `[edit snmp rmon]` hierarchy level to configure RMON history sampling collection parameters. The following parameters are required:

- History index: The history entry is identified by an integer history index value (`historyControlIndex` MIB field) specified when you configure this statement, which is used to display or clear collected results later.
- Interface: The interface to monitor for the specified history index. Only one interface can be associated with a particular RMON history index.

In addition to the required parameters, you can specify a custom sampling `interval` (in seconds) and the sampling `bucket-size` (number of discrete samples to be collected in a given interval).

```
[edit snmp]
user@switch# set rmon history history-index interface interface-name
user@switch# set rmon history history-index interval seconds
user@switch# set rmon history history-index bucket-size number
```

An optional tag (`owner`) associated with the history index can also be assigned to the collection.

View and Clear RMON History Statistics

Use the `show snmp rmon history` command to display collected RMON history table entries. You can also use the `show snmp mib walk` command to view RMON history table field samples.

The following sample RMON configuration sets up a history table sampling for interface xe-0/0/20.0 using a history index value of 1:

```
user@switch# show snmp | display set
set snmp rmon history 1 interface xe-0/0/20.0
set snmp rmon history 1 bucket-size 1000
set snmp rmon history 1 interval 5
set snmp rmon history 1 owner test
```

Using the `show snmp mib walk` command, you can see `etherHistoryPkts` field statistics collected for history index 1:

```
user@switch> show snmp mib walk etherHistoryPkts
etherHistoryPkts.1.1 = 0
<...>
etherHistoryPkts.1.148 = 10
etherHistoryPkts.1.149 = 14
```

To clear collected RMON history statistics, use the `clear snmp history` command. After clearing samples collected up to that point, collection continues again at the configured interval, and new samples are recorded. This command has options to clear collected samples of a particular configured history index or to clear all samples from all configured indices.

For example, the following command clears collected RMON history samples for history control index 1 configured above:

```
user@switch> clear snmp history 1
Samples collected are cleared.

user@switch> show snmp mib walk etherHistoryPkts | no-more


user@switch> show snmp mib walk etherHistoryPkts | no-more
etherHistoryPkts.1.1 = 0
```

Monitor Network Service Quality by using RMON

IN THIS SECTION

- RMON for Monitoring Service Quality | **718**
- Understanding Measurement Points, Key Performance Indicators, and Baseline Values | **723**
- Define and Measure Network Availability | **725**
- Measure Health | **733**
- Measure Performance | **742**

RMON for Monitoring Service Quality

IN THIS SECTION

- Setting Thresholds | **718**
- RMON Command-Line Interface | **720**
- RMON Event Table | **720**
- RMON Alarm Table | **721**
- Troubleshoot RMON | **722**

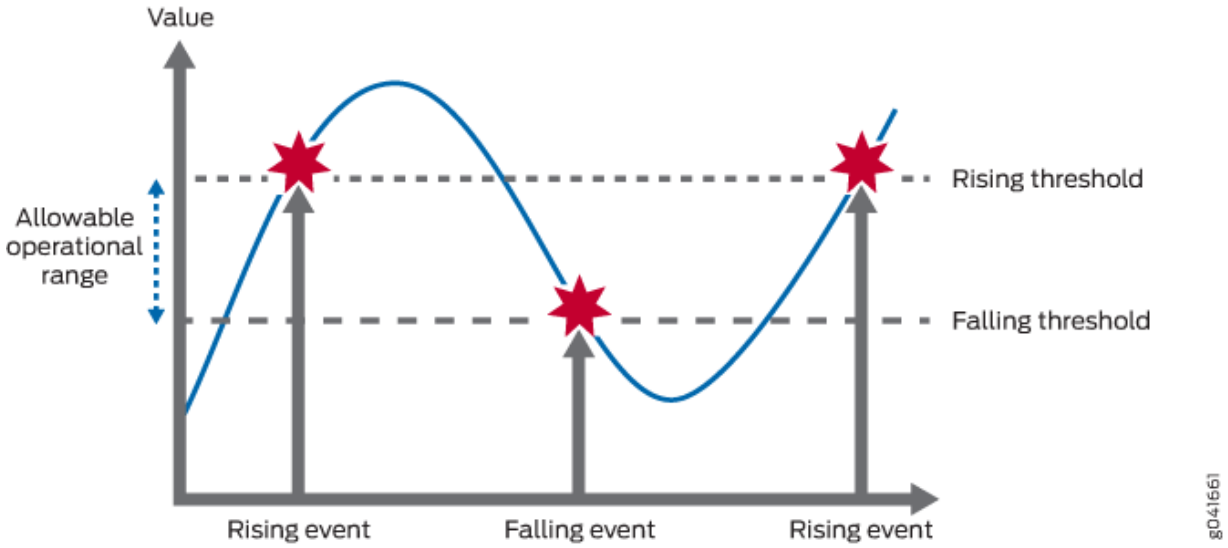
Health and performance monitoring can benefit from the remote monitoring of SNMP variables by the local SNMP agents running on each router. The SNMP agents compare MIB values against predefined thresholds and generate exception alarms without the need for polling by a central SNMP management platform. 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 25 on page 719.)

Figure 25: 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 that 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 63 on page 721.) 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 63: 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 64 on page 721.

Table 64: RMON Alarm Table

Field	Description
alarmStatus	Status of this row (for example, valid, invalid, or createRequest)
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)

Table 64: RMON Alarm Table *(Continued)*

Field	Description
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*.

Troubleshoot 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 65 on page 722 to the RFC 2819 alarmTable.

Table 65: 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.

Understanding Measurement Points, Key Performance Indicators, and Baseline Values

IN THIS SECTION

- Measurement Points | 723
- Basic Key Performance Indicators | 724
- Setting Baselines | 725

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:

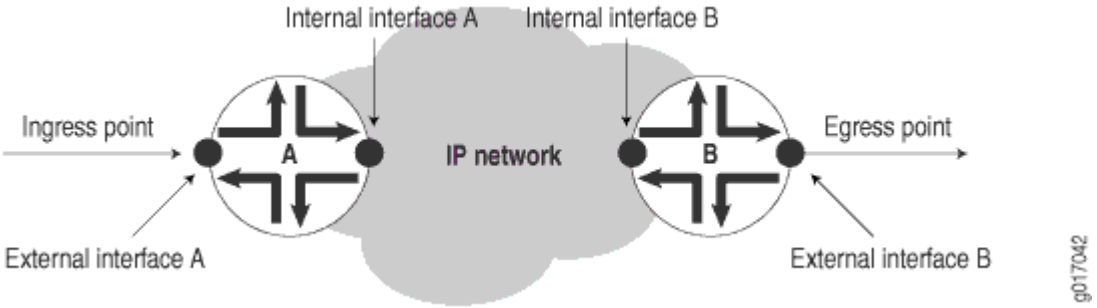
Measurement Points

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 26 on page 724.

- 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 26: Network Entry Points



NOTE: Figure 26 on page 724 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):

- 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.
- 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.
- 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.

Define and Measure Network Availability

IN THIS SECTION

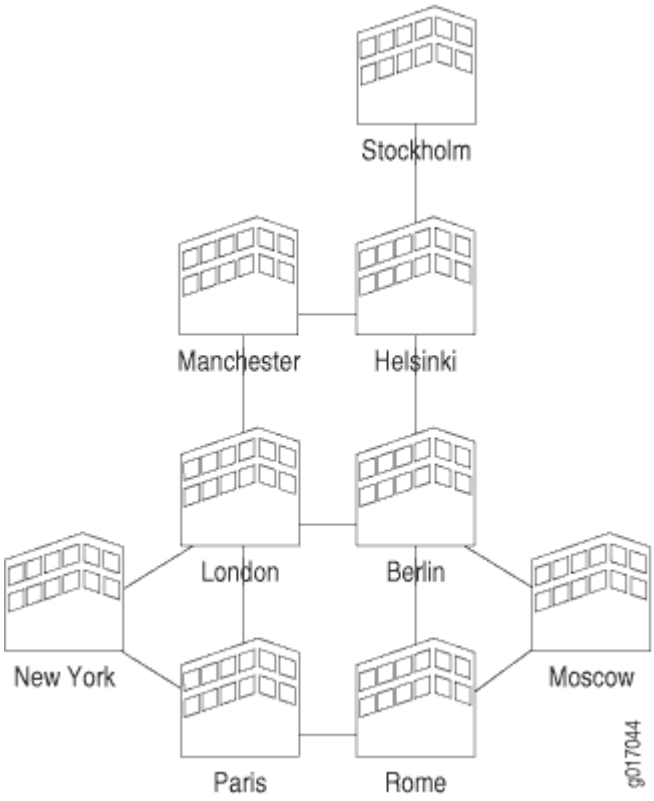
- Define Network Availability | **725**
- Measure Availability | **729**

This topic includes the following sections:

Define Network Availability

Availability of a service provider's IP network can be thought of as the reachability between the regional points of presence (POP), as shown in Figure 27 on page 726.

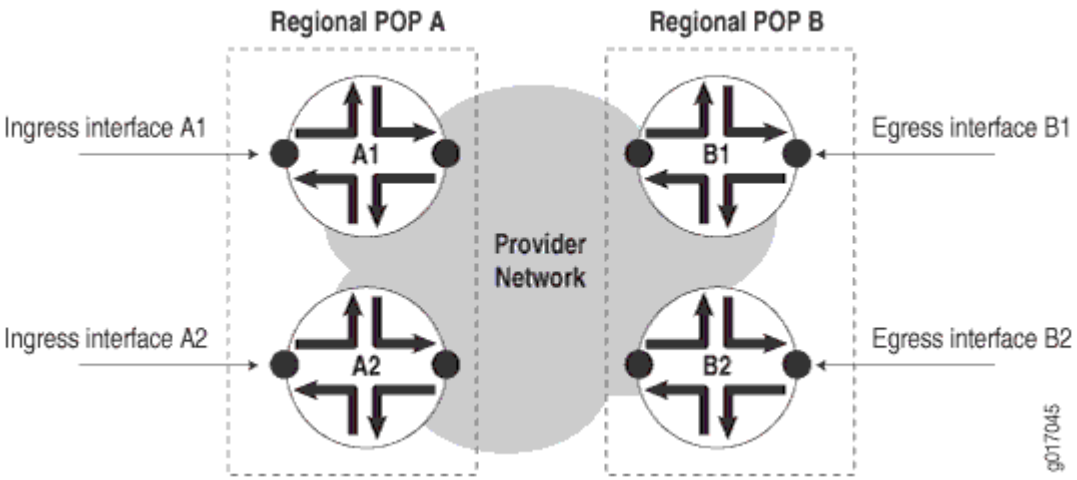
Figure 27: Regional Points of Presence



With the example above, when you use a full mesh of measurement points, where every POP measures the availability to every other POP, you can calculate the total availability of the service provider's network. This KPI can also be used to help monitor the service level of the network, and can be used by the service provider and its customers to determine if they are operating within the terms of their service-level agreement (SLA).

Where a POP may consist of multiple routers, take measurements to each router as shown in Figure 28 on page 727.

Figure 28: Measurements to Each Router



Measurements include:

- Path availability—Availability of an egress interface B1 as seen from an ingress interface A1.
- Router availability—Percentage of path availability of all measured paths terminating on the router.
- POP availability—Percentage of router availability between any two regional POPs, A and B.
- Network availability—Percentage of POP availability for all regional POPs in the service provider's network.

To measure POP availability of POP A to POP B in Figure 28 on page 727, you must measure the following four paths:

```
Path A1 => B1
Path A1 => B2
Path A2 => B1
Path A2 => B2
```

Measuring availability from POP B to POP A would require a further four measurements, and so on.

A full mesh of availability measurements can generate significant management traffic. From the sample diagram above:

- Each POP has two co-located provider edge (PE) routers, each with 2xSTM1 interfaces, for a total of 18 PE routers and 36xSTM1 interfaces.
- There are six core provider (P) routers, four with 2xSTM4 and 3xSTM1 interfaces each, and two with 3xSTM4 and 3xSTM1 interfaces each.

This makes a total of 68 interfaces. A full mesh of paths between every interface is:

[n x (n−1)] / 2 gives [68 x (68−1)] / 2=2278 paths

To reduce management traffic on the service provider's network, instead of generating a full mesh of interface availability tests (for example, from each interface to every other interface), you can measure from each router's loopback address. This reduces the number of availability measurements required to a total of one for each router, or:

[n x (n−1)] / 2 gives [24 x (24−1)] / 2=276 measurements

This measures availability from each router to every other router.

Monitoring the SLA and the Required Bandwidth

A typical SLA between a service provider and a customer might state:

```
A Point of Presence is the connection of two back-to-back provider edge routers to separate core
provider routers using different links for resilience. The system is considered to be
unavailable when either an entire POP becomes unavailable or for the duration of a Priority 1
fault.
```

An SLA availability figure of 99.999 percent for a provider's network would relate to a down time of approximately 5 minutes per year. Therefore, to measure this proactively, you would have to take availability measurements at a granularity of less than one every five minutes. With a standard size of 64 bytes per ICMP ping request, one ping test per minute would generate 7680 bytes of traffic per hour per destination, including ping responses. A full mesh of ping tests to 276 destinations would generate 2,119,680 bytes per hour, which represents the following:

- On an OC3/STM1 link of 155.52 Mbps, a utilization of 1.362 percent
- On an OC12/STM4 link of 622.08 Mbps, a utilization of 0.340 percent

With a size of 1500 bytes per ICMP ping request, one ping test per minute would generate 180,000 bytes per hour per destination, including ping responses. A full mesh of ping tests to 276 destinations would generate 49,680,000 bytes per hour, which represents the following:

- On an OC3/STM1 link, 31.94 percent utilization
- On an OC12/STM4 link, 7.986 percent utilization

Each router can record the results for every destination tested. With one test per minute to each destination, a total of 1 x 60 x 24 x 276 = 397,440 tests per day would be performed and recorded by each router. All ping results are stored in the `pingProbeHistoryTable` (see RFC 2925) and can be retrieved by an SNMP performance reporting application (for example, service performance management software

from InfoVista, Inc., or Concord Communications, Inc.) for post processing. This table has a maximum size of 4,294,967,295 rows, which is more than adequate.

Measure Availability

There are two methods you can use to measure availability:

- Proactive—Availability is automatically measured as often as possible by an operational support system.
- Reactive—Availability is recorded by a Help desk when a fault is first reported by a user or a fault monitoring system.

This section discusses real-time performance monitoring as a proactive monitoring solution.

Real-Time Performance Monitoring

Juniper Networks provides a real-time performance monitoring (RPM) service to monitor real-time network performance. Use the J-Web Quick Configuration feature to configure real-time performance monitoring parameters used in real-time performance monitoring tests. (J-Web Quick Configuration is a browser-based GUI that runs on Juniper Networks routers. For more information, see the *J-Web Interface User Guide*.)

Configuring Real-Time Performance Monitoring

Some of the most common options you can configure for real-time performance monitoring tests are shown in Table 66 on page 729.

Table 66: Real-Time Performance Monitoring Configuration Options

Field	Description
Request Information	

Table 66: Real-Time Performance Monitoring Configuration Options *(Continued)*

Field	Description
Probe Type	Type of probe to send as part of the test. Probe types can be: • http-get • http-get-metadata • icmp-ping • icmp-ping-timestamp • tcp-ping • udp-ping
Interval	Wait time (in seconds) between each probe transmission. The range is 1 to 255 seconds.
Test Interval	Wait time (in seconds) between tests. The range is 0 to 86400 seconds.
Probe Count	Total number of probes sent for each test. The range is 1 to 15 probes.
Destination Port	TCP or UDP port to which probes are sent. Use number 7—a standard TCP or UDP port number—or select a port number from 49152 through 65535.
DSCP Bits	Differentiated Services code point (DSCP) bits. This value must be a valid 6-bit pattern. The default is 000000.
Data Size	Size (in bytes) of the data portion of the ICMP probes. The range is 0 to 65507 bytes.
Data Fill	Contents of the data portion of the ICMP probes. Contents must be a hexadecimal value. The range is 1 to 800h.
Maximum Probe Thresholds	

Table 66: Real-Time Performance Monitoring Configuration Options *(Continued)*

Field	Description
Successive Lost Probes	Total number of probes that must be lost successively to trigger a probe failure and generate a system log message. The range is 0 to 15 probes.
Lost Probes	Total number of probes that must be lost to trigger a probe failure and generate a system log message. The range is 0 to 15 probes.
Round Trip Time	Total round-trip time (in microseconds) from the Services Router to the remote server, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
Jitter	Total *jitter* (in microseconds) for a test, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
Standard Deviation	Maximum allowable standard deviation (in microseconds) for a test, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
Egress Time	Total one-way time (in microseconds) from the router to the remote server, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
Ingress Time	Total one-way time (in microseconds) from the remote server to the router, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
Jitter Egress Time	Total outbound-time jitter (in microseconds) for a test, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
Jitter Ingress Time	Total inbound-time jitter (in microseconds) for a test, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.

Table 66: Real-Time Performance Monitoring Configuration Options *(Continued)*

Field	Description
`Egress Standard Deviation`	Maximum allowable standard deviation of outbound times (in microseconds) for a test, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.
`Ingress Standard Deviation`	Maximum allowable standard deviation of inbound times (in microseconds) for a test, which, if exceeded, triggers a probe failure and generates a system log message. The range is 0 to 60,000,000 microseconds.

Displaying Real-Time Performance Monitoring Information

For each real-time performance monitoring test configured on the router, monitoring information includes the round-trip time, jitter, and standard deviation. To view this information, select `Monitor > RPM` in the J-Web interface, or enter the `show services rpm` command-line interface (CLI) command.

To display the results of the most recent real-time performance monitoring probes, enter the `show services rpm probe-results` CLI command:

```
user@host> show services rpm probe-results
Owner: p1, Test: t1
   Target address: 10.8.4.1, Source address: 10.8.4.2, Probe type: icmp-ping
   Destination interface name: lt-0/0/0.0
   Test size: 10 probes
   Probe results:
      Response received, Sun Jul 10 19:07:34 2005
      Rtt: 50302 usec
   Results over current test:
      Probes sent: 2, Probes received: 1, Loss percentage: 50
      Measurement: Round trip time
        Minimum: 50302 usec, Maximum: 50302 usec, Average: 50302 usec,
        Jitter: 0 usec, Stddev: 0 usec
   Results over all tests:
      Probes sent: 2, Probes received: 1, Loss percentage: 50
      Measurement: Round trip time
        Minimum: 50302 usec, Maximum: 50302 usec, Average: 50302 usec,
        Jitter: 0 usec, Stddev: 0 usec
```

Measure Health

You can monitor health metrics reactively by using fault management software such as SMARTS InCharge, Micromuse Netcool Omnibus, or Concord Live Exceptions. We recommend that you monitor the health metrics shown in Table 67 on page 733.

Table 67: Health Metrics

Metric	Description	Parameters	
		Name	Value
Errors in	Number of inbound packets that contained errors, preventing them from being delivered	**MIB name**	IF-MIB (RFC 2233)
		Variable name	ifInErrors
		Variable OID	.1.3.6.1.31.2.2.1.14
		Frequency (mins)	60
		Allowable range	To be baselined
		Managed objects	Logical interfaces
Errors out	Number of outbound packets that contained errors, preventing them from being transmitted	**MIB name**	IF-MIB (RFC 2233)
		Variable name	ifOutErrors
		Variable OID	.1.3.6.1.31.2.2.1.20
		Frequency (mins)	60
		Allowable range	To be baselined

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Managed objects	Logical interfaces
Discards in	Number of inbound packets discarded, even though no errors were detected	**MIB name**	IF-MIB (RFC 2233)
		Variable name	ifInDiscards
		Variable OID	.1.3.6.1.31.2.2.1.13
		Frequency (mins)	60
		Allowable range	To be baselined
		Managed objects	Logical interfaces
Unknown protocols	Number of inbound packets discarded because they were of an unknown protocol	**MIB name**	IF-MIB (RFC 2233)
		Variable name	ifInUnknownProtos
		Variable OID	.1.3.6.1.31.2.2.1.15
		Frequency (mins)	60
		Allowable range	To be baselined
		Managed objects	Logical interfaces

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
Interface operating status	Operational status of an interface	**MIB name**	IF-MIB (RFC 2233)
		Variable name	ifOperStatus
		Variable OID	.1.3.6.1.31.2.2.1.8
		Frequency (mins)	15
		Allowable range	1 (up)
		Managed objects	Logical interfaces
Label Switched Path (LSP) state	Operational state of an MPLS label-switched path	**MIB name**	MPLS-MIB
		Variable name	mplsLspState
		Variable OID	mplsLspEntry.2
		Frequency (mins)	60
		Allowable range	2 (up)
		Managed objects	All label-switched paths in the network
Component operating status	Operational status of a router hardware component	**MIB name**	JUNIPER-MIB

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Variable name	jnxOperatingState
		Variable OID	.1.3.6.1.4.1.2636.1.13.1.6
		Frequency (mins)	60
		Allowable range	2 (running) or 3 (ready)
		Managed objects	All components in each Juniper Networks router
Component operating temperature	Operational temperature of a hardware component, in Celsius	**MIB name**	JUNIPER-MIB
		Variable name	jnxOperatingTemp
		Variable OID	.1.3.6.1.4.1.2636.1.13.1.7
		Frequency (mins)	60
		Allowable range	To be baselined
		Managed objects	All components in a chassis
System up time	Time, in milliseconds, that the system has been operational.	**MIB name**	MIB-2 (RFC 1213)
		Variable name	sysUpTime

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Variable OID	.1.3.6.1.1.3
		Frequency (mins)	60
		Allowable range	Increasing only (decrement indicates a restart)
		Managed objects	All routers
No IP route errors	Number of packets that could not be delivered because there was no IP route to their destination.	**MIB name**	MIB-2 (RFC 1213)
		Variable name	ipOutNoRoutes
		Variable OID	ip.12
		Frequency (mins)	60
		Allowable range	To be baselined
		Managed objects	Each router
Wrong SNMP community names	Number of incorrect SNMP community names received	**MIB name**	MIB-2 (RFC 1213)
		Variable name	snmpInBadCommunityNames
		Variable OID	snmp.4

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Frequency (mins)	24
		Allowable range	To be baselined
		Managed objects	Each router
SNMP community violations	Number of valid SNMP communities used to attempt invalid operations (for example, attempting to perform SNMP Set requests)	**MIB name**	MIB-2 (RFC 1213)
		Variable name	snmpInBadCommunityUses
		Variable OID	snmp.5
		Frequency (mins)	24
		Allowable range	To be baselined
		Managed objects	Each router
Redundancy switchover	Total number of redundancy switchovers reported by this entity	**MIB name**	JUNIPER-MIB
		Variable name	jnxRedundancySwitchoverCount
		Variable OID	jnxRedundancyEntry.8
		Frequency (mins)	60

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Allowable range	To be baselined
		Managed objects	All Juniper Networks routers with redundant Routing Engines
FRU state	Operational status of each field-replaceable unit (FRU)	**MIB name**	JUNIPER-MIB
		Variable name	jnxFruState
		Variable OID	jnxFruEntry.8
		Frequency (mins)	15
		Allowable range	2 through 6 for ready/online states. See jnxFruOfflineReason in the event of a FRU failure.
		Managed objects	All FRUs in all Juniper Networks routers.
Rate of tail-dropped packets	Rate of tail-dropped packets per output queue, per forwarding class, per interface.	**MIB name**	JUNIPER-COS-MIB
		Variable name	jnxCosIfqTailDropPktRate
		Variable OID	jnxCosIfqStatsEntry.12
		Frequency (mins)	60

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Allowable range	To be baselined
		Managed objects	For each forwarding class per interface in the provider network, when CoS is enabled.
Interface utilization: octets received	Total number of octets received on the interface, including framing characters.	**MIB name**	IF-MIB
		Variable name	ifInOctets
		Variable OID	.1.3.6.1.2.1.2.2.1.10.x
		Frequency (mins)	60
		Allowable range	To be baselined
		Managed objects	All operational interfaces in the network
Interface utilization: octets transmitted	Total number of octets transmitted out of the interface, including framing characters.	**MIB name**	IF-MIB
		Variable name	ifOutOctets
		Variable OID	.1.3.6.1.2.1.2.2.1.16.x
		Frequency (mins)	60

Table 67: Health Metrics *(Continued)*

Metric	Description	Parameters	
		Name	Value
		Allowable range	To be baselined
		Managed objects	All operational interfaces in the network

NOTE: Byte counts vary depending on interface type, encapsulation used and PIC supported. For example, with vlan-ccc encapsulation on a 4xFE, GE, or GE 1Q PIC, the byte count includes framing and control word overhead. (See Table 68 on page 741.)

Table 68: Counter Values for vlan-ccc Encapsulation

PIC Type	Encapsulation	input (Unit Level)	Output (Unit Level)	SNMP
4xFE	vlan-ccc	Frame (no frame check sequence [FCS])	Frame (including FCS and control word)	ifInOctets, ifOutOctets
GE	vlan-ccc	Frame (no FCS)	Frame (including FCS and control word)	ifInOctets, ifOutOctets
GE IQ	vlan-ccc	Frame (no FCS)	Frame (including FCS and control word)	ifInOctets, ifOutOctets

SNMP traps are also a good mechanism to use for health management. For more information, see "'SNMP Traps Supported by Junos OS" on page 467" and "Enterprise-Specific SNMP Traps Supported by Junos OS."

Measure Performance

IN THIS SECTION

- Measure Class of Service | 745
- Inbound Firewall Filter Counters per Class | 747
- Monitor Output Bytes per Queue | 748
- Calculate Dropped Traffic | 749

The performance of a service provider's network is usually defined as how well it can support services, and is measured with metrics such as delay and utilization. We suggest that you monitor the following performance metrics using applications such as InfoVista Service Performance Management or Concord Network Health (see Table 69 on page 742).

Table 69: Performance Metrics

Metric:	Average delay
Description	Average round-trip time (in milliseconds) between two measurement points.
MIB name	DISMAN-PING-MIB (RFC 2925)
Variable name	`pingResultsAverageRtt`
Variable OID	pingResultsEntry.6
Frequency (mins)	15 (or depending upon ping test frequency)
Allowable range	To be baselined
Managed objects	Each measured path in the network
Metric:	Interface utilization

Description	Utilization percentage of a logical connection.
MIB name	IF-MIB
Variable name	(ifInOctets & ifOutOctets) * 8 / ifSpeed
Variable OID	ifTable entries
Frequency (mins)	60
Allowable range	To be baselined
Managed objects	All operational interfaces in the network
Metric:	Disk utilization
Description	Utilization of disk space within the Juniper Networks router
MIB name	HOST-RESOURCES-MIB (RFC 2790)
Variable name	**hrStorageSize – hrStorageUsed**
Variable OID	hrStorageEntry.5 – hrStorageEntry.6
Frequency (mins)	1440
Allowable range	To be baselined
Managed objects	All Routing Engine hard disks
Metric:	Memory utilization
Description	Utilization of memory on the Routing Engine and FPC.

MIB name	JUNIPER-MIB (Juniper Networks enterprise Chassis MIB)
Variable name	**jnxOperatingHeap**
Variable OID	Table for each component
Frequency (mins)	60
Allowable range	To be baselined
Managed objects	All Juniper Networks routers
Metric:	CPU load
Description	Average utilization over the past minute of a CPU.
MIB name	JUNIPER-MIB (Juniper Networks enterprise Chassis MIB)
Variable name	**jnxOperatingCPU**
Variable OID	Table for each component
Frequency (mins)	60
Allowable range	To be baselined
Managed objects	All Juniper Networks routers
Metric:	LSP utilization
Description	Utilization of the MPLS label-switched path.
MIB name	MPLS-MIB

Variable name	**mplsPathBandwidth / (mplsLspOctets * 8)**
Variable OID	mplsLspEntry.21 and mplsLspEntry.3
Frequency (mins)	60
Allowable range	To be baselined
Managed objects	All label-switched paths in the network
Metric:	Output queue size
Description	Size, in packets, of each output queue per forwarding class, per interface.
MIB name	JUNIPER-COS-MIB
Variable name	**jnxCosIfqQedPkts**
Variable OID	jnxCosIfqStatsEntry.3
Frequency (mins)	60
Allowable range	To be baselined
Managed objects	For each forwarding class per interface in the network, once CoS is enabled.

This section includes the following topics:

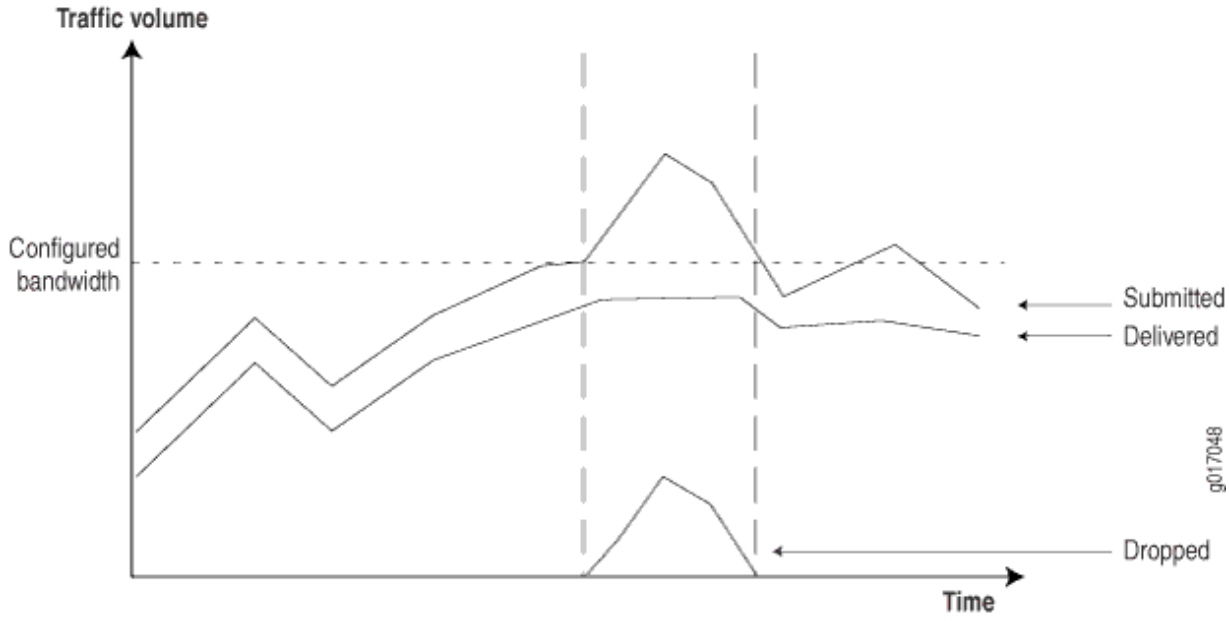
Measure Class of Service

You can use class-of-service (CoS) mechanisms to regulate how certain classes of packets are handled within your network during times of peak congestion. Typically you must perform the following steps when implementing a CoS mechanism:

- Identify the type of packets that is applied to this class. For example, include all customer traffic from a specific ingress edge interface within one class, or include all packets of a particular protocol such as voice over IP (VoIP).
- Identify the required deterministic behavior for each class. For example, if VoIP is important, give VoIP traffic the highest priority during times of network congestion. Conversely, you can downgrade the importance of Web traffic during congestion, as it may not impact customers too much.

With this information, you can configure mechanisms at the network ingress to monitor, mark, and police traffic classes. Marked traffic can then be handled in a more deterministic way at egress interfaces, typically by applying different queuing mechanisms for each class during times of network congestion. You can collect information from the network to provide customers with reports showing how the network is behaving during times of congestion. (See Figure 29 on page 746.)

Figure 29: Network Behavior During Congestion



To generate these reports, routers must provide the following information:

- Submitted traffic—Amount of traffic received per class.
- Delivered traffic—Amount of traffic transmitted per class.
- Dropped traffic—Amount of traffic dropped because of CoS limits.

The following section outlines how this information is provided by Juniper Networks routers.

Inbound Firewall Filter Counters per Class

Firewall filter counters are a very flexible mechanism you can use to match and count inbound traffic per class, per interface. For example:

```
firewall {
    filter f1 {
        term t1 {
            from {
                dscp af11;
            }
            then {
                # Assured forwarding class 1 drop profile 1 count inbound-af11;
                accept;
            }
        }
    }
}
```

For example, Table 70 on page 747 shows additional filters used to match the other classes.

Table 70: Inbound Traffic Per Class

DSCP Value	Firewall Match Condition	Description
10	af11	Assured forwarding class 1 drop profile 1
12	af12	Assured forwarding class 1 drop profile 2
18	af21	Best effort class 2 drop profile 1
20	af22	Best effort class 2 drop profile 2
26	af31	Best effort class 3 drop profile 1

Any packet with a CoS DiffServ code point (DSCP) conforming to RFC 2474 can be counted in this way. The Juniper Networks enterprise-specific Firewall Filter MIB presents the counter information in the variables shown in Table 71 on page 748.

Table 71: Inbound Counters

Indicator Name	Inbound Counters
MIB	jnxFirewalls
Table	jnxFirewallCounterTable
Index	jnxFWFilter.jnxFWCounter
Variables	jnxFWCounterPacketCount jnxFWCounterByteCount
Description	Number of bytes being counted pertaining to the specified firewall filter counter
SNMP version	SNMPv2

This information can be collected by any SNMP management application that supports SNMPv2. Products from vendors such as Concord Communications, Inc., and InfoVista, Inc., provide support for the Juniper Networks Firewall MIB with their native Juniper Networks device drivers.

Monitor Output Bytes per Queue

You can use the Juniper Networks enterprise ATM CoS MIB to monitor outbound traffic, per virtual circuit forwarding class, per interface. (See Table 72 on page 748.)

Table 72: Outbound Counters for ATM Interfaces

Indicator Name	Outbound Counters
MIB	JUNIPER-ATM-COS-MIB
Variable	jnxCosAtmVcQstatsOutBytes
Index	ifIndex.atmVclVpi.atmVclVci.jnxCosFcId

Table 72: Outbound Counters for ATM Interfaces *(Continued)*

Indicator Name	Outbound Counters
Description	Number of bytes belonging to the specified forwarding class that were transmitted on the specified virtual circuit.
SNMP version	SNMPv2

Non-ATM interface counters are provided by the Juniper Networks enterprise-specific CoS MIB, which provides information shown in Table 73 on page 749.

Table 73: Outbound Counters for Non-ATM Interfaces

Indicator Name	Outbound Counters
MIB	`JUNIPER-COS-MIB`
Table	`jnxCosIfqStatsTable`
Index	`jnxCosIfqIfIndex.jnxCosIfqFc`
Variables	`jnxCosIfqTxedBytes` `jnxCosIfqTxedPkts`
Description	Number of transmitted bytes or packets per interface per forwarding class
SNMP version	SNMPv2

Calculate Dropped Traffic

You can calculate the amount of dropped traffic by subtracting the outbound traffic from the incoming traffic:

```
Dropped = Inbound Counter - Outbound Counter
```

You can also select counters from the CoS MIB, as shown in Table 74 on page 750.

Table 74: Dropped Traffic Counters

Indicator Name	Dropped Traffic
MIB	JUNIPER-COS-MIB
Table	jnxCosIfqStatsTable
Index	jnxCosIfqIfIndex.jnxCosIfqFc
Variables	jnxCosIfqTailDropPkts jnxCosIfqTotalRedDropPkts
Description	The number of tail-dropped or RED-dropped packets per interface per forwarding class
SNMP version	SNMPv2

Health Monitoring with SNMP

IN THIS SECTION

- Health Monitoring Overview | **751**
- Configure Health Monitoring on Devices Running Junos OS | **752**
- Configure Health Monitoring | **756**

Health Monitoring Overview

Health monitoring is an SNMP feature that extends the RMON alarm infrastructure to provide monitoring for a predefined set of objects (such as file system usage, CPU usage, and memory usage), and for Junos OS processes.

You enable the health monitor feature using the `health-monitor` statement at the `[edit snmp]` hierarchy level. You can also configure health monitor parameters such as a falling threshold, rising threshold, and interval. If the value of a monitored object exceeds the rising or falling threshold, an alarm is triggered and an event may be logged.

The falling threshold is the lower threshold for the monitored object instance. The rising threshold is the upper threshold for the monitored object instance. Each threshold is expressed as a percentage of the maximum possible value. The interval represents the period of time, in seconds, over which the object instance is sampled and compared with the rising and falling thresholds.

Events are only generated when a threshold is first crossed in any one direction, rather than after each sample interval. For example, if a rising threshold alarm, along with its corresponding event, is raised, no more threshold crossing events occur until a corresponding falling alarm occurs.

System log entries for health monitor events have a corresponding HEALTHMONITOR tag and not a generic SNMPD_RMON_EVENTLOG tag. However, the health monitor sends generic RMON risingThreshold and fallingThreshold traps. You can use the `show snmp health-monitor` operational command to view information about health monitor alarms and logs.

When you configure the health monitor, monitoring information for certain object instances is available, as shown in Table 75 on page 751.

Table 75: Monitored Object Instances

Object	Description
jnxHrStoragePercentUsed.1	Monitors the **/dev/ad0s1a:** file system on the switch. This is the root file system mounted on /.
jnxHrStoragePercentUsed.2	Monitors the **/dev/ad0s1e:** file system on the switch. This is the configuration file system mounted on **/config**.
jnxOperatingCPU (RE0)	Monitors CPU usage by the Routing Engine (RE0).
jnxOperatingBuffer (RE0)	Monitors the amount of memory available on the Routing Engine (RE0).

Table 75: Monitored Object Instances *(Continued)*

Object	Description
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.

SEE ALSO

- *falling-threshold (Health Monitor)*
- *interval (Health Monitor)*
- *rising-threshold (Health Monitor)*
- show snmp health-monitor

Configure Health Monitoring on Devices Running Junos OS

IN THIS SECTION

- Monitored Objects | **753**
- Minimum Health Monitoring Configuration | **755**
- Configure the Falling Threshold or Rising Threshold | **755**
- Configure the Interval | **756**
- Log Entries and Traps | **756**

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;
    idp {
        falling-threshold percentage;
        interval seconds;
        rising-threshold percentage;
    }
}
```

Configuring monitoring events at the `[edit snmp health-monitor]` hierarchy level sets polling intervals for the overall system health. If you set these same options at the `[edit snmp health-monitor idp]` hierarchy level, an SNMP event is generated by the device if the percentage of dataplane memory utilized by the intrusion detection and prevention (IDP) system rises above or falls below your settings.

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 76 on page 754.

Table 76: 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)	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.
jnxOperatingCPU (RE1)	
jnxOperatingBuffer (RE0)	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.
jnxOperatingBuffer (RE1)	
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;
```

Configure 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.

Configure 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.

SEE ALSO

> *health-monitor*

Configure Health Monitoring

This topic describes how to configure the health monitor feature for QFX Series devices.

The health monitor feature extends the SNMP RMON alarm infrastructure to provide predefined monitoring for a selected set of object instances (such as file system usage, CPU usage, and memory usage) and dynamic object instances (such as Junos OS processes).

In this procedure, 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.

To configure health monitoring:

1. Configure the health monitor:

```
[edit snmp]
user@switch# set health-monitor
```

2. Configure the falling threshold:

```
[edit snmp]
user@switch# set health-monitor falling-threshold percentage
```

For example:

```
user@switch# set health-monitor falling-threshold 85
```

3. Configure the rising threshold:

```
[edit snmp]
user@switch# set health-monitor rising-threshold percentage
```

For example:

```
user@switch# set health-monitor rising-threshold 75
```

4. Configure the interval:

```
[edit snmp]
user@switch# set health-monitor interval seconds
```

For example:

```
user@switch# set health-monitor interval 600
```

SEE ALSO

falling-threshold

interval (Health Monitor)

rising-threshold (Health Monitor)

5

PART

Accounting Options

Accounting Options Overview | 760

Configure Accounting Options, Source Class Usage and Destination Class Usage Options | 761

Accounting Options Overview

An accounting profile represents common characteristics of collected accounting data, including the following:

- Collection interval
- File to contain accounting data
- Specific fields and counter names on which to collect statistics

You can configure multiple accounting profiles, as described in Table 77 on page 760.

Table 77: Types of Accounting Profiles

Type of Profile	Description
Interface profile	Collects the specified error and statistic information.
Filter profile	Collects the byte and packet counts for the counter names specified in the filter profile.
MIB profile	Collects selected MIB statistics and logs them to a specified file.
Routing Engine profile	Collects selected Routing Engine statistics and logs them to a specified file.
Class usage profile	Collects class usage statistics and logs them to a specified file.

Configure Accounting Options, Source Class Usage and Destination Class Usage Options

IN THIS SECTION

- Configuration Statements at the [edit accounting-options] Hierarchy Level | 761
- Accounting Options Configuration | 763
- Configure Accounting-Data Log Files | 773
- Manage Accounting Files | 779
- Configure the Interface Profile | 780
- Configure the Filter Profile | 784
- Example: Configure a Filter Profile | 786
- Example: Configure Interface-Specific Firewall Counters and Filter Profiles | 787
- Configure Class Usage Profiles | 789
- Configure the MIB Profile | 792
- Configure the Routing Engine Profile | 795

Configuration Statements at the [edit accounting-options] Hierarchy Level

This topic shows all possible configuration statements at the `[edit accounting-options]` 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]
accounting-options {
    class-usage-profile profile-name {
        file filename;
        interval minutes;
        destination-classes {
            destination-class-name;
        }
```

```
        source-classes {
            source-class-name;
        }
    }
    file filename {
        archive-sites {
        }
        files number;
        nonpersistent;
        size bytes;
        start-time time;
        transfer-interval minutes;
    }
    filter-profile profile-name {
        counters {
            counter-name;
        }
        file filename;
        interval minutes;
    }
}
interface-profile profile-name {
    fields {
        field-name;
    }
    file filename;
    interval minutes;
}
mib-profile profile-name {
    file filename;
    interval seconds;
    object-names {
        mib-object-name;
    }
    operation operation-name;
}
routing-engine-profile profile-name {
    fields {
        field-name;
    }
    file filename;
```

```
    interval minutes;
}
```

Accounting Options Configuration

IN THIS SECTION

- Accounting Options—Full Configuration | 763
- Minimum Accounting Options Configuration | 768

This topic contains the following sections:

Accounting Options—Full Configuration

To configure accounting options, include the following statements at the [edit accounting-options] hierarchy level:

```
accounting-options {
    class-usage-profile profile-name {
        file filename;
        interval minutes;
        destination-classes {
            destination-class-name;
        }
        source-classes {
            source-class-name;
        }
        file filename {
            archive-sites {
                site-name;
            }
            files number;
            nonpersistent;
            size bytes;
            source-classes time;
```

```
            transfer-interval minutes;
        }
        filter-profile profile-name {
            counters {
                counter-name;
            }
            file filename;
            interval minutes;
        }
    }
    flat-file-profile profile-name{
        fields {
            all-fields;
            egress-stats {
                all-fields;
                input-bytes;
                input-packets;
                output-bytes;
                output-packets;
                queue-id;
                red-drop-bytes;
                red-drop-packets;
                tail-drop-packets;
                total-drop-packets;
            }
            general-param {
                all-fields;
                accounting-type;
                descr;
                line-id;
                logical-interface;
                nas-port-id;
                physical-interface;
                routing-instance;
                timestamp;
                vlan-id;
            }
            ingress-stats {
                all-fields;
                drop-packets;
                input-bytes;
                input-packets;
                output-bytes;
```

```
            output-packets;
            queue-id;
        }
        l2-stats {
            all-fields;
            input-mcast-bytes;
            input-mcast-packets;
        }
        fields {
            all-fields;
            egress-stats {
                all-fields;
                input-bytes;
                input-packets;
                output-bytes;
                output-packets;
                queue-id;
                red-drop-bytes;
                red-drop-packets;
                tail-drop-packets;
                total-drop-packets;
            }
            general-param {
                all-fields;
                accounting-type;
                descr;
                line-id;
                logical-interface;
                nas-port-id;
                physical-interface;
                routing-instance;
                timestamp;
                vlan-id;
            }
            ingress-stats {
                all-fields;
                drop-packets;
                input-bytes;
                input-packets;
                output-bytes;
                output-packets;
                queue-id;
            }
```

```
general-param {
    all-fields;
    accounting-type;
    descr;
    line-id;
    logical-interface;
    nas-port-id;
    physical-interface;
    routing-instance;
    timestamp;
    vlan-id;
}
ingress-stats {
    all-fields;
    drop-packets;
    input-bytes;
    input-packets;
    output-bytes;
    output-packets;
    queue-id;
}
l2-stats {
    all-fields;
    input-mcast-bytes;
    input-mcast-packets;
}
overall-packet {
    all-fields;
    input-bytes;
    input-discards;
    input-errors;
    input-packets;
    inputv6-bytes;
    inputv6-packets;
    output-bytes;
    output-errors;
    output-packets;
    outputv6-bytes;
    outputv6-packets;
    input-v4-bytes;
    input-v4-packets;
    output-v4-bytes;
    output-v4-packets;
```

```
                    input-bytes-per-sec;
                    input-packets-per-sec;
                }
            }
            file filename;
            format (csv | ipdr)
            interval minutes;
             schema-version schema-name;
        }
        interface-profile profile-name {
            fields {
                field-name;
            }
            file filename;
            interval minutes;
        }
        mib-profile profile-name {
            file filename;
            interval (Accounting Options) seconds;
            object-names {
                mib-object-name;
            }
            operation operation-name;
        }
        routing-engine-profile profile-name {
            fields {
                field-name;
            }
            file filename;
            interval minutes;
        }
    }
}
```

By default, accounting options are disabled.

NOTE: Do not configure MIB objects related to interface octets or packets for a MIB profile, because doing so can cause the SNMP walk or a CLI show command to time out.

Minimum Accounting Options Configuration

To enable accounting options on the router, you must perform at least the following tasks:

- Configure accounting options by including a `file` statement and one or more `source-class-usage`, `destination-class-profile`, `filter-profile`, `interface-profile`, `mib-profile`, or `routing-engine-profile` statements at the `[edit accounting-options]` hierarchy level:

```
[edit]
accounting-options {
    class-usage-profile profile-name {
        file filename;
        interval minutes;
        source-classes {
            source-class-name;
        }
        destination-classes {
            destination-class-name;
        }
        file filename {
            archive-sites {
                site-name;
            }
            files number;
            size bytes;
            transfer-interval minutes;
        }
        filter-profile profile-name {
            counters {
                counter-name;
            }
            file filename;
            interval minutes;
        }
        flat-file-profile profile-name{
            fields {
                all-fields;
                egress-stats {
                    all-fields;
                    input-bytes;
                    input-packets;
                    output-bytes;
```

```
        output-packets;
        queue-id;
        red-drop-bytes;
        red-drop-packets;
        tail-drop-packets;
        total-drop-packets;
    }
    general-param {
        all-fields;
        accounting-type;
        descr;
        line-id;
        logical-interface;
        nas-port-id;
        physical-interface;
        routing-instance;
        timestamp;
        vlan-id;
    }
    ingress-stats {
        all-fields;
        drop-packets;
        input-bytes;
        input-packets;
        output-bytes;
        output-packets;
        queue-id;
    }
    l2-stats {
        all-fields;
        input-mcast-bytes;
        input-mcast-packets;
    }
    overall-packet {
        all-fields;
        input-bytes;
        input-discards;
        input-errors;
        input-packets;
        inputv6-bytes;
        inputv6-packets;
        output-bytes;
        output-errors;
```

```
                output-packets;
                outputv6-bytes;
                outputv6-packets;
                input-v4-bytes;
                input-v4-packets;
                output-v4-bytes;
                output-v4-packets;
                input-bytes-per-sec;
                input-packets-per-sec;
            }
        }
        file filename;
        format (csv | ipdr)
        interval minutes;
         schema-version schema-name;
    }
    flat-file-profile profile-name{
        fields {
            all-fields;
            egress-stats {
                all-fields;
                input-bytes;
                input-packets;
                output-bytes;
                output-packets;
                queue-id;
                red-drop-bytes;
                red-drop-packets;
                tail-drop-packets;
                total-drop-packets;
            }
            general-param {
                all-fields;
                accounting-type;
                descr;
                line-id;
                logical-interface;
                nas-port-id;
                physical-interface;
                routing-instance;
                timestamp;
                vlan-id;
            }
```

```
                ingress-stats {
                    all-fields;
                    drop-packets;
                    input-bytes;
                    input-packets;
                    output-bytes;
                    output-packets;
                    queue-id;
                }
                l2-stats {
                    all-fields;
                    input-mcast-bytes;
                    input-mcast-packets;
                }
                overall-packet {
                    all-fields;
                    input-bytes;
                    input-discards;
                    input-errors;
                    input-packets;
                    inputv6-bytes;
                    inputv6-packets;
                    output-bytes;
                    output-errors;
                    output-packets;
                    outputv6-bytes;
                    outputv6-packets;
                    input-v4-bytes;
                    input-v4-packets;
                    output-v4-bytes;
                    output-v4-packets;
                    input-bytes-per-sec;
                    input-packets-per-sec;
                }
            }
            file filename;
            format (csv | ipdr)
            interval minutes;
             schema-version schema-name;
        }
        interface-profile profile-name {
            fields {
                field-name;
```

```
            }
            file filename;
            interval minutes;
        }
        mib-profile profile-name {
            file filename;
            interval minutes;
            object-names {
                mib-object-name;
            }
            operation operation-name;
        }
        routing-engine-profile profile-name {
            fields {
                field-name;
            }
            file filename;
            interval minutes;
        }
    }
}
```

- Apply the profiles to the chosen interfaces or filters.

  Apply an interface profile to a physical or logical interface by including the `accounting-profile` statement at either the `[edit interfaces interface-name]` or the `[edit interfaces interface-name unit logical-unit-number]` hierarchy level.

```
[edit interfaces]
interface-name {
    accounting-profile profile-name;
    unit logical-unit-number {
        accounting-profile profile-name;
    }
}
```

NOTE: You do not apply destination class profiles to interfaces. Although the interface needs to have the `destination-class-usage` statement configured, the destination class profile automatically finds all interfaces with the destination class configured.

Apply a filter profile to a firewall filter by including the `accounting-profile` statement at the `[edit firewall filter filter-name]` hierarchy level:

```
[edit firewall]
filter filter-name {
    accounting-profile profile-name;
}
```

You do not need to apply the Routing Engine profile to an interface because the statistics are collected on the Routing Engine itself.

Configure Accounting-Data Log Files

IN THIS SECTION

- Configure How Long Backup Files Are Retained | **774**
- Configure the Maximum Size of the File | **775**
- Configure Archive Sites for the Files | **775**
- Configure Local Backup for Accounting Files | **776**
- Configure Files to Be Compressed | **776**
- Configure the Maximum Number of Files | **777**
- Configure the Storage Location of the File | **777**
- Configure Files to Be Saved After a Change in Primary Role | **777**
- Configure the Start Time for File Transfer | **778**
- Configure the Transfer Interval of the File | **778**

An accounting profile specifies what statistics to collect and write to a log file. To configure an accounting-data log file, include the `file` statement at the `[edit accounting-options]` hierarchy level:

```
[edit accounting-options]
cleanup-interval {
    interval days;
}
```

```
file filename {
    archive-sites {
        site-name;
    }
    backup-on-failure  (master-and-slave | master-only);
    files number;
    nonpersistent;
    push-backup-to-master;
    size bytes;
    start-time time;
    transfer-interval minutes;
}
```

where *filename* is the name of the file in which to write accounting data.

If the filename contains spaces, enclose it in quotation marks (" "). The filename cannot contain a forward slash (/). The file is created in the `/var/log` directory and can contain data from multiple profiles.

All accounting-data log files include header and trailer sections that start with a `#` in the first column. The header contains the file creation time, the hostname, and the columns that appear in the file. The trailer contains the time that the file was closed.

Whenever any configured value changes that affects the columns in a file, the file creates a new profile layout record that contains a new list of columns.

You must configure the file size; all other properties are optional.

Configure How Long Backup Files Are Retained

You can configure how many days the files are retained in the local directory before they are deleted.

NOTE: Files saved to the **/var/log/pfedBackup** directory are always compressed to conserve local storage, regardless of whether the `compress` statement is configured.

To configure retention for backup files:

- Specify the number of days.

```
[edit accounting-options]
user@host# set cleanup-interval interval days
```

NOTE: Files are retained for 1 day if you do not configure this option.

This value, whether configured or default, applies to all configured files at the `[edit accounting-options file]` hierarchy level.

Configure the Maximum Size of the File

To configure the maximum size of the file:

- Specify the size.

```
[edit accounting-options file filename]
size bytes;
```

The `size` statement is the maximum size of the log file, in bytes, kilobytes (KB), megabytes (MB), or gigabytes (GB). The minimum value for *bytes* is 256 KB. You must configure *bytes*; the remaining attributes are optional.

Configure Archive Sites for the Files

After a file reaches its maximum size or the `transfer-interval` time is exceeded, the file is closed, renamed, and, if you configured an archive site, transferred to a remote host.

To configure the sites where files are archived:

- Specify one or more site names.

```
[edit accounting-options file filename]
user@host# set archive-sites site-name
```

where *site-name* is any valid FTP URL. For more information about specifying valid FTP URLs, see the Junos OS Administration Library. You can specify more than one URL, in any order. When a file is archived, the router or switch attempts to transfer the file to the first URL in the list, trying the next site in the list only if the transfer does not succeed. The log file is stored at the archive site with a filename of the format *router-name_log-filename_timestamp*. When you configure file archival by using `archive-states` statement, the transfer file utility uses the default routing instance to connect to the destination server. If the default routing instance is unable to connect to the destination server, the transfer file utility does not work.

Starting in Junos OS 18.4R1, when you configure file archival by using the `archive-sites` statement, the transfer file utility does not work if you have enabled the management instance.

Configure Local Backup for Accounting Files

You can configure the router to save a copy of the accounting file locally when the normal transfer of the files to the archive site fails. The file is saved to the **/var/log/pfedBackup** directory of the relevant Routing Engine. You must specify whether only the files from the primary Routing Engine are saved or files are saved from both the primary Routing Engine and the backup (client) Routing Engine.

NOTE: Files saved to the **/var/log/pfedBackup** directory are always compressed to conserve local storage, regardless of whether the `compress` statement is configured.

To configure local backup in the event of failure:

- Specify local backup and which files are saved.

```
[edit accounting-options file filename]
user@host# set backup-on-failure  (master-and-slave | master-only)
```

Disabling this feature deletes the backed-up accounting files from the directory.

NOTE: When you do not configure this option, the file is saved on failure into the local directory specified as the last site in the list of archive sites.

Configure Files to Be Compressed

By default, accounting files are transferred in an uncompressed format. To conserve resources during transmission and on the archive site, you can configure compression for the files.

NOTE: Files saved to the **/var/log/pfedBackup** directory are always compressed to conserve local storage, regardless of whether the `compress` statement is configured.

To configure the router to compress accounting files when they are transferred:

- Specify compression.

```
[edit accounting-options file filename]
user@host# set compress
```

Configure the Maximum Number of Files

To configure the maximum number of files:

- Specify the number.

```
[edit accounting-options file filename]
user@host# set files number
```

When a log file reaches its maximum size, it is renamed *filename*.0, then *filename*.1, and so on, until the maximum number of log files is reached. Then the oldest log file is overwritten. The minimum value for *number* is 3 and the default value is 10.

Configure the Storage Location of the File

On J Series Services Routers, the files are stored by default on the compact flash drive. Alternatively, you can configure the files to be stored in the **mfs/var/log** directory (on DRAM) instead of the cf/var/log directory (on the compact flash drive).

To configure the storage location on DRAM:

- Specify nonpersistent storage.

```
[edit accounting-options file filename]
user@host# set nonpersistent
```

This feature is useful for minimizing read/write traffic on the router's compact flash drive.

NOTE: If log files for accounting data are stored on DRAM, these files are lost when you reboot the router. We recommend that you back up these files periodically.

Configure Files to Be Saved After a Change in Primary Role

You can configure the router to save the accounting files from the new backup Routing Engine to the new primary Routing Engine when a change in primary role occurs. The files are stored in the **/var/log/pfedBackup** directory on the router. The primary Routing Engine includes these accounting files with its own current accounting files when it transfers the files from the backup directory to the archive site at the next transfer interval. Configure this option when the new backup Routing Engine is not able to connect to the archive site; for example, when the site is not connected by means of an out-of-band interface or the path to the site is routed through a line card.

To configure the backup Routing Engine files to be saved when primary role changes:

- Specify the backup.

```
[edit accounting-options file filename]
user@host# set push-backup-to-master
```

NOTE: The backup Routing Engine's files on the primary Routing Engine are sent at each interval even though the files remain the same. If this is more activity than you want, consider using the `backup-on-failure master-and-slave` statement instead.

Configure the Start Time for File Transfer

To configure the start time for transferring files:

- Specify the time.

```
[edit accounting-options file filename]
user@host# set start-time YYYY-MM-DD.hh:mm
```

For example, 10:00 a.m. on January 30, 2007 is represented as `2007-01-30.10:00`.

Configure the Transfer Interval of the File

To configure the interval at which files are transferred:

- Specify the interval.

```
[edit accounting-options file filename]
user@host# set transfer-interval minutes
```

The range for `transfer-interval` is 5 through 2880 minutes. The default is 30 minutes.

TIP: Junos OS saves the existing log file and creates a new file at the configured transfer intervals irrespective of whether:

- The file has reached the maximum size.
- An archive site is configured.

When you have a relatively small transfer interval configured and if no archive site is configured, data can be lost as Junos OS overwrites the log files when the maximum number of log files is reached. To ensure that the log information is saved for a reasonably long time:

- Configure an archive site to archive the log files every time a new log file is created.
- Configure the maximum value (2880 minutes) for `transfer-interval` so that new files are created less frequently; that is, only when the file exceeds the maximum size limit or once in 2 days.

Manage Accounting Files

If you configure SRX300, SRX320, SRX340, SRX345, SRX550M, SRX1500, SRX4100, SRX4200, and SRX4600 devices to capture accounting data in log files, set the location for your accounting files to the DRAM.

The default location for accounting files is the `cfs/var/log` directory on the CompactFlash (CF) card. The `nonpersistent` option minimizes the read/write traffic to your CF card. We recommend that you use the `nonpersistent` option for all accounting files configured on your system.

To store accounting log files in DRAM instead of the CF card:

1. Enter configuration mode in the CLI.
2. Create an accounting data log file in DRAM and replace *filename* with the name of the file.

```
[edit]
user@host# edit accounting-options file filename
```

3. Store accounting log files in the DRAM file.

```
[edit]
user@host# set file filename nonpersistent
```

CAUTION: If log files for accounting data are stored on DRAM, these files are lost when the device reboots. Therefore, we recommend that you back up these files periodically.

NOTE: The CLI `nonpersistent` option is not supported on SRX5000 line.

Configure the Interface Profile

IN THIS SECTION

- Configure Fields | **781**
- Configure the File Information | **781**
- Configure Cleared Statistics to be Reported in the Flat File | **781**
- Configure the Interval | **782**
- Example: Configure the Interface Profile | **782**

An interface profile specifies the information collected and written to a log file. You can configure a profile to collect error and statistic information for input and output packets on a particular physical or logical interface.

To configure an interface profile, include the `interface-profile` statement at the `[edit accounting-options]` hierarchy level:

```
[edit accounting-options]
interface-profile profile-name {
    fields {
        field-name;
    }
    file filename;
    interval minutes;
}
```

By default, the Packet Forwarding Engine (PFE) periodically collects the statistics for all interfaces. To improve the performance, you can optionally disable the periodic refresh by including the `periodic-refresh disable` statement at the `[edit accounting-options]` hierarchy level.

Each accounting profile must have a unique *profile-name*. To apply a profile to a physical or logical interface, include the `accounting-profile` statement at either the `[edit interfaces` *interface-name*`]` or the `[edit interfaces` *interface-name* `unit` *logical-unit-number*`]` hierarchy level. You can also apply an accounting profile at the `[edit firewall family` *family-type* `filter` *filter-name*`]` hierarchy level. For more information, see the Routing Policies, Firewall Filters, and Traffic Policers User Guide.

To configure an interface profile, perform the tasks described in the following sections:

Configure Fields

An interface profile must specify what statistics are collected. To configure which statistics should be collected for an interface, include the `fields` statement at the `[edit accounting-options interface-profile` *profile-name*`]` hierarchy level:

```
[edit accounting-options interface-profile profile-name]
fields {
    field-name;
}
```

Configure the File Information

Each accounting profile logs its statistics to a file in the **/var/log** directory.

To configure which file to use, include the `file` statement at the `[edit accounting-options interface-profile` *profile-name*`]` hierarchy level:

```
[edit accounting-options interface-profile profile-name]
file filename;
```

You must specify a `file` statement for the interface profile that has already been configured at the `[edit accounting-options]` hierarchy level.

Configure Cleared Statistics to be Reported in the Flat File

When you issue the `clear interfaces statistics` command for a logical interface configured to collect accounting statistics, all accounting statistics received on that interface from the Packet Forwarding Engine are cleared. The current values when the command is issued become the new baseline and the statistics counters are reset to zero. The new values, starting from zero, are displayed in the CLI. However, they are not reported that way in the accounting flat file associated with the interface. Instead, the values as reported in the file continue to increment as if the command had not been issued.

You can change this result by including the `allow-clear` statement in the interface profile. In this case, when you issue the `clear interfaces statistics` command, the statistics are reset to zero and reported to the flat file.

To configure reporting of cleared accounting statistics to the flat file, specify reporting:

```
[edit accounting-options interface-profile profile-name]
 allow-clear;
```

Configure the Interval

Each interface with an accounting profile enabled has statistics collected once per interval time specified for the accounting profile. Statistics collection time is scheduled evenly over the configured interval. To configure the interval, include the `interval` statement at the `[edit accounting-options interface-profile profile-name]` hierarchy level:

```
[edit accounting-options interface-profile profile-name]
interval minutes;
```

NOTE: The minimum interval allowed is 1 minute. Configuring a low interval in an accounting profile for a large number of interfaces might cause serious performance degradation.

The range for the `interval` statement is 1 through 2880 minutes. The default is 30 minutes.

Example: Configure the Interface Profile

Configure the interface profile:

```
[edit]
accounting-options {
    file if_stats {
        size 40 files 5;
    }
    interface-profile if_profile1 {
        file if_stats;
        interval 30;
        fields {
            input-bytes;
            output-bytes;
```

```
            input-packets;
            output-packets;
            input-multicast;
            output-multicast;
        }
    }
    interface-profile if_profile2 {
        file if_stats;
        interval 30;
        fields {
            input-bytes;
            output-bytes;
            input-packets;
            output-packets;
            input-multicast;
            output-multicast;
        }
    }
    interfaces {
        xe-1/0/0 {
            accounting-profile if_profile1;
            unit 0 {
                accounting-profile if_profile2;
                ...
            }
        }
    }
}
```

The two interface profiles, if-profile1 and if-profile2, write data to the same file, if-stats. The if-stats file might look like the following:

```
#FILE CREATED 976823478 2000-12-14-19:51:18
#hostname host
#profile-layout if_profile2,epoch-timestamp,interface-name,snmp-index,input-bytes,output-bytes,
input-packets,output-packets,input-multicast,output-multicast
#profile-layout if_profile1,epoch-timestamp,interface-name,snmp-index,input-bytes,output-bytes,
input-packets
if_profile2,976823538,xe-1/0/0.0,8,134696815,3681534,501088,40723,0,0
if_profile1,976823538,xe-1/0/0,7,134696815,3681534,501088
```

```
...
#FILE CLOSED 976824378 2000-12-14-20:06:18
```

Configure the Filter Profile

IN THIS SECTION

- Configure the Counters | 785
- Configure the File Information | 785
- Configure the Interval | 785

A filter profile specifies error and statistics information collected and written to a file. A filter profile must specify counter names for which statistics are collected.

To configure a filter profile, include the `filter-profile` statement at the `[edit accounting-options]` hierarchy level:

```
[edit accounting-options]
filter-profile profile-name {
    counters {
        counter-name;
    }
    file filename;
    interval minutes;
}
```

To apply the filter profile, include the `accounting-profile` statement at the `[edit firewall filter filter-name]` hierarchy level.

To configure a filter profile, perform the tasks described in the following sections:

Configure the Counters

Statistics are collected for all counters specified in the filter profile. To configure the counters, include the `counters` statement at the `[edit accounting-options filter-profile profile-name]` hierarchy level:

```
[edit accounting-options filter-profile profile-name]
counters {
}
```

Configure the File Information

Each accounting profile logs its statistics to a file in the `/var/log` directory.

To configure which file to use, include the `file` statement at the `[edit accounting-options filter-profile profile-name]` hierarchy level:

```
[edit accounting-options filter-profile profile-name]
file filename;
```

You must specify a filename for the filter profile that has already been configured at the `[edit accounting-options]` hierarchy level.

NOTE: The limit on the total number of characters per line in a log file equals 1023. If this limit is exceeded, the output written to the log file is incomplete. Ensure that you limit the number of counters or requested data so that this character limit is not exceeded.

NOTE: If the configured file size or transfer interval is exceeded, Junos OS closes the file and starts a new one. By default, the transfer interval value is 30 minutes. If the transfer interval is not configured, Junos OS closes the file and starts a new one when the file size exceeds its configured value or the default transfer interval value exceeds 30 minutes. To avoid transferring files every 30 minutes, specify a different value for the transfer interval.

Configure the Interval

Each filter with an accounting profile enabled has statistics collected once per interval time specified for the accounting profile. Statistics collection time is scheduled evenly over the configured interval. To

configure the interval, include the `interval` statement at the `[edit accounting-options filter-profile profile-name]` hierarchy level:

```
[edit accounting-options filter-profile profile-name]
interval;
```

NOTE: The minimum interval allowed is 1 minute. Configuring a low interval in an accounting profile for a large number of filters might cause serious performance degradation.

The range for the `interval` statement is 1 through 2880 minutes. The default is 30 minutes.

Example: Configure a Filter Profile

Configure a filter profile:

```
[edit]
accounting-options {
    file fw_accounting {
        size 500k files 4;
    }
    filter-profile fw_profile1 {
        file fw_accounting;
        interval 60;
        counters {
            counter1;
            counter2;
            counter3;
        }
    }
}
firewall {
    filter myfilter {
        accounting-profile fw_profile1;
        ...
        term accept-all {
            then {
                count counter1;
```

```
                accept;
            }
        }
    }
}
```

The filter profile, `fw-profile1`, writes data to the file `fw_accounting`. The file might look like the following:

```
#FILE CREATED 976825278 2000-12-14-20:21:18
#hostname host
#profile-layout fw_profile1,epoch-timestamp,filter-name,counter-name,packet-count,byte-count
fw_profile1,976826058,myfilter,counter1,163,10764
...
#FILE CLOSED 976826178 2000-12-14-20:36:18
```

Example: Configure Interface-Specific Firewall Counters and Filter Profiles

To collect and log count statistics collected by firewall filters on a per-interface basis, you must configure a filter profile and include the interface-specific statement at the `[edit firewall filter` *`filter-name`*`]` hierarchy level.

Configure the firewall filter accounting profile:

```
[edit accounting-options]
file cust1_accounting {
    size 500k;
}
filter-profile cust1_profile {
    file cust1_accounting;
    interval 1;
    counters {
        r1;
    }
}
```

Configure the interface-specific firewall counter:

```
[edit firewall]
filter f3 {
    accounting-profile cust1_profile;
    interface-specific;
    term f3-term {
        then {
            count r1;
            accept;
        }
    }
}
```

Apply the firewall filter to an interface:

```
[edit interfaces]
xe-1/0/0 {
    unit 0 {
        family inet {
            filter {
                input f3;
                output f3;
            }
            address 20.20.20.30/24;
        }
    }
}
```

The following example shows the contents of the `cust1_accounting` file in the `/var/log` folder that might result from the preceding configuration:

```
#FILE CREATED 995495212 2001-07-18-22:26:52
#hostname host
#profile-layout cust1_profile,epoch-timestamp,interfaces,filter-name,
counter-name,packet-count,byte-count
cust1_profile,995495572,xe-1/0/0.0,f3-xe-1/0/0.0-i,r1-xe-1/0/0.0-i,5953,1008257
cust1_profile,995495602,xe-1/0/0.0,f3-xe-1/0/0.0-o,r1-xe-1/0/0.0-o,5929,1006481
...
```

If the interface-specific statement is not included in the configuration, the following output might result:

```
#FILE CREATED 995495212 2001-07-18-22:26:52
#hostname host
#profile-layout cust1_profile,epoch-timestamp,interfaces,filter-name,
counter-name,packet-count,byte-count
cust1_profile,995495572,xe-1/0/0.0,f3,r1,5953,1008257
cust1_profile,995495632,xe-1/0/0.0,f3,r1,5929,1006481
```

Configure Class Usage Profiles

IN THIS SECTION

- Configure a Class Usage Profile | **789**
- Configure the File Information | **790**
- Configure the Interval | **790**
- Create a Class Usage Profile to Collect Source Class Usage Statistics | **790**
- Create a Class Usage Profile to Collect Destination Class Usage Statistics | **791**

To collect class usage statistics, perform the tasks described in these sections:

Configure a Class Usage Profile

You can configure the class usage profile to collect statistics for particular source and destination classes.

To configure the class usage profile to filter by source classes, include the source-classes statement at the [edit accounting-options class-usage-profile *profile-name*] hierarchy level:

```
[edit accounting-options class-usage-profile profile-name]
source-classes {
    source-class-name;
}
```

To configure the class usage profile to filter by destination classes, include the `destination-classes` statement at the `[edit accounting-options class-usage-profile profile-name]` hierarchy level:

```
[edit accounting-options class-usage-profile profile-name]
destination-classes {
    destination-class-name;
}
```

Configure the File Information

Each accounting profile logs its statistics to a file in the `/var/log` directory.

To specify which file to use, include the `file` statement at the `[edit accounting-options class-usage-profile profile-name]` hierarchy level:

```
[edit accounting-options class-usage-profile profile-name]
file filename;
```

You must specify a filename for the source class usage profile that has already been configured at the `[edit accounting-options]` hierarchy level. You can also specify a filename for the destination class usage profile configured at the `[edit accounting-options]` hierarchy level.

Configure the Interval

Each interface with a class usage profile enabled has statistics collected once per interval specified for the accounting profile. Statistics collection time is scheduled evenly over the configured interval. To configure the interval, include the `interval` statement at the `[edit accounting-options class-usage-profile profile-name]` hierarchy level:

```
[edit accounting-options class-usage-profile profile-name]
interval;
```

Create a Class Usage Profile to Collect Source Class Usage Statistics

To create a class usage profile to collect source class usage statistics:

```
[edit]
accounting-options {
    class-usage-profile scu-profile1;
    file usage-stats;
```

```
    interval 15;
    source-classes {
        gold;
        silver;
        bronze;
    }
}
```

The class usage profile, `scu-profile1`, writes data to the file `usage_stats`. The file might look like the following:

```
#FILE CREATED 976825278 2000-12-14-20:21:18
#profile-layout, scu_profile,epoch-timestamp,interface-name,source-class,
packet-count,byte-count
scu_profile,980313078,xe-1/0/0.0,gold,82,6888
scu_profile,980313078,xe-1/0/0.0,silver,164,13776
scu_profile,980313078,xe-1/0/0.0,bronze,0,0
scu_profile,980313678,xe-1/0/0.0,gold,82,6888
scu_profile,980313678,xe-1/0/0.0,silver,246,20664
scu_profile,980313678,xe-1/0/0.0,bronze,0,0
```

Create a Class Usage Profile to Collect Destination Class Usage Statistics

To create a class usage profile to collect destination class usage statistics:

```
[edit]
accounting-options {
    class-usage-profile dcu-profile1;
    file usage-stats
    interval 15;
    destination-classes {
        gold;
        silver;
        bronze;
    }
}
```

The class usage profile, dcu-profile1, writes data to the file usage-stats. The file might look like the following:

```
#FILE CREATED 976825278 2000-12-14-20:21:18
#profile-layout, dcu_profile,epoch-timestamp,interface-name,destination-class,
packet-count,byte-count
dcu_profile,980313078,xe-1/0/0.0,gold,82,6888
dcu_profile,980313078,xe-1/0/0.0,silver,164,13776
dcu_profile,980313078,xe-1/0/0.0,bronze,0,0
dcu_profile,980313678,xe-1/0/0.0,gold,82,6888
dcu_profile,980313678,xe-1/0/0.0,silver,246,20664
dcu_profile,980313678,xe-1/0/0.0,bronze,0,0
...
#FILE CLOSED 976826178 2000-12-14-20:36:18
```

Configure the MIB Profile

IN THIS SECTION

- Configure the File Information | 793
- Configure the Interval | 793
- Configure the MIB Operation | 793
- Configure MIB Object Names | 794
- Example: Configure a MIB Profile | 794

The MIB profile collects MIB statistics and logs them to a file. The MIB profile specifies the SNMP operation and MIB object names for which statistics are collected.

To configure a MIB profile, include the mib-profile statement at the [edit accounting-options] hierarchy level:

```
[edit accounting-options]
mib-profile profile-name {
    file filename;
```

```
    interval minutes;
    object-names {
        mib-object-name;
    }
    operation operation-name;
}
```

To configure a MIB profile, perform the tasks described in the following sections:

Configure the File Information

Each accounting profile logs its statistics to a file in the `/var/log` directory.

To configure which file to use, include the `file` statement at the `[edit accounting-options mib-profile profile-name]` hierarchy level:

```
[edit accounting-options mib-profile profile-name]
file filename;
```

You must specify a *filename* for the MIB profile that has already been configured at the `[edit accounting-options]` hierarchy level.

Configure the Interval

A MIB profile has statistics collected once per interval time specified for the profile. Statistics collection time is scheduled evenly over the configured interval. To configure the interval, include the `interval` statement at the `[edit accounting-options mib-profile profile-name]` hierarchy level:

```
[edit accounting-options mib-profile profile-name]
interval;
```

The range for the `interval` statement is 1 through 2880 minutes. The default is 30 minutes.

Configure the MIB Operation

A MIB profile must specify the operation that is used to collect MIB statistics. To configure which operation is used to collect MIB statistics, include the `operation` statement at the `[edit accounting-options mib-profile profile-name]` hierarchy level:

```
[edit accounting-options mib-profile profile-name]
operation operation-name;
```

You can configure a `get`, `get-next`, or `walk` operation. The default operation is `walk`.

Configure MIB Object Names

A MIB profile must specify the MIB objects for which statistics are to be collected. To configure the MIB objects for which statistics are collected, include the `objects-names` statement at the `[edit accounting-options mib-profile profile-name]` hierarchy level:

```
[edit accounting-options mib-profile profile-name]
object-names {
    mib-object-name;
}
```

You can include multiple MIB object names in the configuration.

NOTE: In Junos OS Release 15.1X49-D10 and later, do not configure MIB objects related to interface octets or packets for a MIB profile, because it can cause the SNMP walk or a CLI show command to time out.

Example: Configure a MIB Profile

Configure a MIB profile:

```
[edit accounting-options]
mib-profile mstatistics {
    file stats;
    interval 60;
    operation walk;
    objects-names {
        ipCidrRouteStatus;
    }
}
```

Configure the Routing Engine Profile

IN THIS SECTION

- Configure Fields | 795
- Configure the File Information | 796
- Configure the Interval | 796
- Example: Configure a Routing Engine Profile | 796

The Routing Engine profile collects Routing Engine statistics and logs them to a file. The Routing Engine profile specifies the fields for which statistics are collected.

To configure a Routing Engine profile, include the `routing-engine-profile` statement at the `[edit accounting-options]` hierarchy level:

```
[edit accounting-options]
routing-engine-profile profile-name {
    fields {
        field-name;
    }
    file filename;
    interval minutes;
}
```

To configure a Routing Engine profile, perform the tasks described in the following sections:

Configure Fields

A Routing Engine profile must specify what statistics are collected. To configure which statistics should be collected for the Routing Engine, include the `fields` statement at the `[edit accounting-options routing-engine-profile profile-name]` hierarchy level:

```
[edit accounting-options routing-engine-profile profile-name]
fields {
    field-name;
}
```

Configure the File Information

Each accounting profile logs its statistics to a file in the `/var/log` directory.

To configure which file to use, include the `file` statement at the `[edit accounting-options routing-engine-profile profile-name]` hierarchy level:

```
[edit accounting-options routing-engine-profile profile-name]
file filename;
```

You must specify a *filename* for the Routing Engine profile that has already been configured at the `[edit accounting-options]` hierarchy level.

Configure the Interval

A Routing Engine profile has statistics collected once per interval time specified for the profile. Statistics collection time is scheduled evenly over the configured interval. To configure the interval, include the `interval` statement at the `[edit accounting-options routing-engine-profile profile-name]` hierarchy level:

```
[edit accounting-options routing-engine-profile profile-name]
interval;
```

The range for `interval` is 1 through 2880 minutes. The default is 30 minutes.

Example: Configure a Routing Engine Profile

Configure a Routing Engine profile:

```
[edit accounting-options]
file my-file {
    size 300k;
}
routing-engine-profile profile-1 {
    file my-file;
    fields {
        host-name;
        date;
        time-of-day;
        uptime;
        cpu-load-1;
        cpu-load-5;
        cpu-load-15;
```

```
    }
}
```

Release History Table

Release	Description
18.4R1	Starting in Junos OS 18.4R1, when you configure file archival by using the `archive-sites` statement, the transfer file utility does not work if you have enabled the management instance.
15.1X49-D10	In Junos OS Release 15.1X49-D10 and later, do not configure MIB objects related to interface octets or packets for a MIB profile, because it can cause the SNMP walk or a CLI show command to time out.

6

PART

Monitoring Options

Interface Alarms | 799

IP Monitoring | 811

sFlow Monitoring Technology | 831

Adaptive Sampling for Routers and Switches | 858

Packet Flow Accelerator Diagnostics Software | 862

CHAPTER 7

Interface Alarms

IN THIS CHAPTER

- Alarm Overview | 799
- Example: Configure Interface Alarms | 806

Alarm Overview

SUMMARY

This section describes interface alarms and how to configure them.

IN THIS SECTION

- Alarm Types | 799
- Alarm Severity | 800
- Alarm Conditions | 801

Alarms alert you to conditions on a network interface, on the device chassis, or in the system software that might prevent the device from operating normally. You can set the conditions that trigger alarms on an interface. Chassis and system alarm conditions are preset.

An active alarm lights the **ALARM** LED on the front panel of the device. You can monitor active alarms from the J-Web user interface or the CLI. When an alarm condition triggers an alarm, the device lights the yellow (amber) **ALARM** LED on the front panel. When the condition is corrected, the light turns off.

Alarm Types

The device supports three types of alarms:

- Interface alarms indicate a problem in the state of the physical links on fixed or installed Physical Interface Modules (PIMs). To enable interface alarms, you must configure them.
- Chassis alarms indicate a failure on the device or one of its components. Chassis alarms are preset and cannot be modified.

- System alarms indicate a missing rescue configuration or software license, where valid. System alarms are preset and cannot be modified, although you can configure them to appear automatically in the J-Web user interface or CLI.

Starting with Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, a new system alarm is introduced to indicate that the PICs (I/O card or SPC) have failed to come online during system start time.

Starting in Junos OS Releases 12.3X48-D85, 15.1X49-D180, and 19.2R1, a system alarm is triggered when the Network Security Process (NSD) is unable to restart due to the failure of one or more NSD subcomponents. The alarm logs about the NSD are saved in the messages log. The alarm is automatically cleared when NSD restarts successfully. The `show chassis alarms` and `show system alarms` commands are updated to display the following output when NSD is unable to restart - `NSD fails to restart because subcomponents fail`.

NOTE: Run the following commands when the CLI prompt indicates that an alarm has been raised:

- `show system alarms`
- `show chassis alarms`
- `show chassis fpc pic-status`

For more information about the CLI commands, see show system alarms, *show chassis alarms*, and show chassis fpc.

Alarm Severity

Alarms have two severity levels:

- Major (red)—Indicates a critical situation on the device that has resulted from one of the following conditions. A red alarm condition requires immediate action.
  - One or more hardware components have failed.
  - One or more hardware components have exceeded temperature thresholds.
  - An alarm condition configured on an interface has triggered a critical warning.
- Minor (yellow)—Indicates a noncritical condition on the device that, if left unchecked, might cause an interruption in service or degradation in performance. A yellow alarm condition requires monitoring or maintenance.

  A missing rescue configuration or software license generates a yellow system alarm.

Alarm Conditions

To enable alarms on a device interface, you must select an alarm condition and an alarm severity. In contrast, alarm conditions and severity are preconfigured for chassis alarms and system alarms.

NOTE: For information about chassis alarms for your device, see the Hardware Guide for your device.

This section contains the following topics:

Interface Alarm Conditions

Table 78 on page 801 lists the interface conditions, sorted by interface type, that you can configure for an alarm. You can configure each alarm condition to trigger either a major (red) alarm or minor a (yellow) alarm. The corresponding configuration option is included.

For the services stateful firewall filters (NAT, IDP, and IPsec), which operate on an internal adaptive services module within a device, you can configure alarm conditions on the integrated services and services interfaces.

Table 78: Interface Alarm Conditions

Interface	Alarm Condition	Description	Configuration Option
DS1 (T1)	Alarm indication signal (AIS)	The normal T1 traffic signal contained a defect condition and has been replaced by the AIS. A transmission interruption occurred at the remote endpoint or upstream of the remote endpoint. This all-ones signal is transmitted to prevent consequential downstream failures or alarms.	**ais**
	Yellow alarm	The remote endpoint is in yellow alarm failure. This condition is also known as a far-end alarm failure.	**ylw**
Ethernet	Link is down	The physical link is unavailable.	**link-down**

Table 78: Interface Alarm Conditions *(Continued)*

Interface	Alarm Condition	Description	Configuration Option
Integrated services	Hardware or software failure	On the adaptive services module, either the hardware associated with the module or the software that drives the module has failed.	**failure**
Serial	Clear-to-send (CTS) signal absent	The remote endpoint of the serial link is not transmitting a CTS signal. The CTS signal must be present before data can be transmitted across a serial link.	**cts-absent**
	Data carrier detect (DCD) signal absent	The remote endpoint of the serial link is not transmitting a DCD signal. Because the DCD signal transmits the state of the device, no signal probably indicates that the remote endpoint of the serial link is unavailable.	**dcd-absent**
	Data set ready (DSR) signal absent	The remote endpoint of the serial link is not transmitting a DSR signal. The DSR signal indicates that the remote endpoint is ready to receive and transmit data across the serial link.	**dsr-absent**
	Loss of receive clock	The clock signal from the remote endpoint is not present. Serial connections require clock signals to be transmitted from one endpoint and received by the other endpoint of the link.	**loss-of-rx-clock**
	Loss of transmit clock	The local clock signal is not present. Serial connections require clock signals to be transmitted from one endpoint and received by the other endpoint of the link.	**loss-of-tx-clock**
Services	Services module hardware down	A hardware problem has occurred on the device's services module. This error typically means that one or more of the CPUs on the module has failed.	**hw-down**

Table 78: Interface Alarm Conditions *(Continued)*

Interface	Alarm Condition	Description	Configuration Option
	Services link down	The link between the device and its services module is unavailable.	**linkdown**
	Services module held in reset	The device's services module is stuck in reset mode. If the services module fails to start up five or more times in a row, the services module is held in reset mode. Startup fails when the amount of time from CPU release to CPU halt is less than 300 seconds.	**pic-hold-reset**
	Services module reset	The device's services module is resetting. The module resets after it crashes or is reset from the CLI, or when it takes longer than 60 seconds to start up.	**pic-reset**
	Services module software down	A software problem has occurred on the device's services module.	**sw-down**
E3	Alarm indication signal (AIS)	The normal E3 traffic signal contained a defect condition and has been replaced by the AIS. A transmission interruption occurred at the remote endpoint or upstream of the remote endpoint. This all-ones signal is transmitted to prevent consequential downstream failures or alarms.	**ais**
	Loss of signal (LOS)	No remote E3 signal is being received at the E3 interface.	**los**
	Out of frame (OOF)	An OOF condition has existed for 10 seconds. This alarm applies only to E3 interfaces configured in frame mode. The OOF failure is cleared when no OOF or LOS defects have occurred for 20 seconds.	**oof**

Table 78: Interface Alarm Conditions *(Continued)*

Interface	Alarm Condition	Description	Configuration Option
	Remote defect indication	An AIS, LOS, or OOF condition exists. This alarm applies only to E3 interfaces configured in frame mode.	**rdi**
T3 (DS3)	Alarm indication signal	The normal T3 traffic signal contained a defect condition and has been replaced by the AIS. A transmission interruption occurred at the remote endpoint or upstream of the remote endpoint. This all-ones signal is transmitted to prevent consequential downstream failures or alarms.	**ais**
	Excessive number of zeros	The bit stream received from the upstream host has more consecutive zeros than are allowed in a T3 frame.	**exz**
	Far-end receive failure (FERF)	The remote endpoint of the connection has failed. A FERF differs from a yellow alarm, because the failure can be any failure, not just an OOF or LOS failure.	**ferf**
	Idle alarm	The Idle signal is being received from the remote endpoint.	**idle**
	Line code violation	Either the line encoding along the T3 link is corrupted or a mismatch between the encoding at the local and remote endpoints of a T3 connection occurred.	**lcv**
	Loss of frame (LOF)	An OOF or loss-of-signal LOS condition has existed for 10 seconds. The LOF failure is cleared when no OOF or LOS defects have occurred for 20 seconds. A LOF failure is also called a red failure.	**lof**

Table 78: Interface Alarm Conditions *(Continued)*

Interface	Alarm Condition	Description	Configuration Option
	Loss of signal (LOS)	No remote T3 signal is being received at the T3 interface.	**los**
	Phase-locked loop out of lock	The clocking signals for the local and remote endpoints no longer operate in lock-step.	**pll**
	Yellow alarm	The remote endpoint is in yellow alarm failure. This condition is also known as a far-end alarm failure.	**ylw**

System Alarm Conditions

Table 79 on page 805 lists the two preset system alarms, the condition that triggers each alarm, and the action you take to correct the condition.

Table 79: System Alarm Conditions and Corrective Actions

Alarm Type	Alarm Condition	Corrective Action
Configuration	The rescue configuration is not set.	Set the rescue configuration.
License	You have configured at least one software feature that requires a feature license, but no valid license for the feature is currently installed. **NOTE**: This alarm indicates that you are in violation of the software license agreement. You must install a valid license key to be in compliance with all agreements.	Install a valid license key.

Release History Table

Release	Description
15.1X49-D60	Starting with Junos OS Release 15.1X49-D60 and Junos OS Release 17.3R1, a new system alarm is introduced to indicate that the PICs (I/O card or SPC) have failed to come online during system start time.
12.3X48-D85 15.1X49-D180 19.2R1	Starting in Junos OS Releases 12.3X48-D85, 15.1X49-D180, and 19.2R1, a system alarm is triggered when the Network Security Process (NSD) is unable to restart due to the failure of one or more NSD subcomponents. The alarm logs about the NSD are saved in the messages log. The alarm is automatically cleared when NSD restarts successfully. The `show chassis alarms` and `show system alarms` commands are updated to display the following output when NSD is unable to restart - `NSD fails to restart because subcomponents fail`.

Example: Configure Interface Alarms

IN THIS SECTION

- Requirements | **806**
- Overview | **807**
- Configuration | **807**
- Verification | **810**

This example shows how to configure interface alarms.

Requirements

Before you begin:

- Establish basic connectivity.
- Configure network interfaces. See Interfaces User Guide for Security Devices.
- Select the network interface on which to apply an alarm and the condition you want to trigger the alarm.

Overview

In this example, you enable interface alarms by explicitly setting alarm conditions. You configure the system to generate a red interface alarm when a yellow alarm is detected on a DS1 link. You configure the system to generate a red interface alarm when a link-down failure is detected on an Ethernet link.

For a serial link, you set cts-absent and dcd-absent to yellow to signify either the CST or the DCD signal is not detected. You set loss-of-rx-clock and loss-of-tx-clock to red alarm to signify either the receiver clock signal or the transmission clock signal is not detected.

For a T3 link, you set the interface alarm to red when the remote endpoint is experiencing a failure. You set exz to yellow alarm when the upstream bit has more consecutive zeros than are permitted in a T3 interface. You then set a red alarm when there is loss-of-signal on the interface.

Finally, you configure the system to display active system alarms whenever a user with the login class admin logs in to the device.

Configuration

IN THIS SECTION

- Procedure | 807

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set chassis alarm ds1 ylw red
set chassis alarm ethernet link-down red
set chassis alarm serial cts-absent yellow dcd-absent yellow
set chassis alarm serial loss-of-rx-clock red loss-of-tx-clock red
set chassis alarm t3 ylw red exz yellow los red
set system login class admin login-alarms
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure interface alarms:

1. Configure an alarm.

```
[edit]
user@host# edit chassis alarm
```

2. Specify the interface alarms on a DS1 and an Ethernet link.

```
[edit chassis alarm]
user@host# set ds1 ylw red
user@host# set ethernet link-down red
```

3. Specify the interface alarms on a serial link.

```
[edit chassis alarm]
user@host# set serial cts-absent yellow
user@host# set serial dcd-absent yellow
user@host# set serial loss-of-rx-clock red
user@host# set serial loss-of-tx-clock red
```

4. Specify the interface alarms on a T3 link.

```
[edit chassis alarm]
user@host# set t3 ylw red
user@host# set t3 exz yellow
user@host# set t3 los red
```

5. Configure the system to display active system alarms.

```
[edit]
user@host# edit system login
user@host# set class admin login-alarms
```

Results

From configuration mode, confirm your configuration by entering the `show chassis alarms` and `show system login` commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
    user@host# show chassis alarms
    t3 {
    exz yellow;
        los red;
    ylw red;
    }
    ds1 {
        ylw red;
    }
    ethernet {
        link-down red;
    }
    serial {
        loss-of-rx-clock red;
        loss-of-tx-clock red;
        dcd-absent yellow;
        cts-absent yellow;
    }
    [edit]
        user@host# show system login
        show system login
    show system login
}
```

If you are done configuring the device, enter `commit` from configuration mode.

Verification

IN THIS SECTION

- Verifying the Alarm Configurations | **810**

Verifying the Alarm Configurations

Purpose

Confirm that the configuration is working properly.

Verify that the alarms are configured.

Action

From configuration mode, enter the `show chassis alarms` command. Verify that the output shows the intended configuration of the alarms.

CHAPTER 8

IP Monitoring

IN THIS CHAPTER

- IP Monitoring Overview | 811
- Example: Configure IP Monitoring on SRX Series Firewalls | 814
- Example: Configure IP Monitoring on SRX5000 line | 818
- Example: Configure Chassis Cluster Redundancy Group IP Address Monitoring | 826

IP Monitoring Overview

SUMMARY

This section describes how to keep track of the status of the system in use.

IN THIS SECTION

- IP Monitoring Test Parameters | 812
- IP Monitoring Through Redundant Ethernet Interface Link Aggregation Groups | 813

This feature monitors IP on standalone SRX Series Firewalls or a *chassis cluster* redundant Ethernet (reth) interface. Existing RPM probes are sent to an IP address to check for reachability. The user takes action based on the reachability result. Supported action currently is preferred static route injection to system route table.

The actions supported are:

- Adding or deleting a new static route that has a higher priority (lower preference) value than a route configured through the CLI command `set routing-options static route`
- Defining multiple probe names under the same IP monitoring policy. If any probe fails, the action is taken. If all probes are reachable, the action is reverted
- Configuring multiple tests in one RPM probe. All tests must fail for the RPM probe to be considered unreachable. If at least one test reaches its target, the RPM probe is considered reachable

- Configuring multiple failure thresholds in one RPM test. If one threshold is reached, the test fails. If no thresholds are reached, the test succeeds.
- Specifying the no-preempt option. If the no-preempt option is specified, the policy does not perform preemptive failback when it is in a failover state or when the RPM probe test recovers from a failure.
- Setting preferred metric values. If the preferred metric value is set, during failover, the route is injected with the set preferred metric value.
- Enabling and disabling interfaces.
    - **Interface-Enable**—On a physical or *logical interface*, when the interface-enable action is configured, the initial state of the interface is disable after startup, and it continues to remain in the disable state as long as the associated RPM probe is in the pass state. When the associated RPM probe fails, the configured physical and logical interfaces are enabled.
    - **Interface-Disable**—On a physical or logical interface, when the interface-disable action is configured, the interface state remains unchanged. When the associated RPM probe fails, the physical and logical interfaces are disabled.

NOTE: Multiple probe names and actions can be defined for the same IP monitoring policy.

IP Monitoring Test Parameters

Each probed target is monitored over the course of a test, which represents a collection of probes during which statistics such as standard deviation and *jitter* are collected are calculated. During a test, probes are generated and responses collected at a rate defined by the probe interval, the number of seconds between probes.

NOTE: To avoid flap, an action is reverted only at the end of a test cycle. During the test cycle, if no threshold is reached, the action is reverted. Although action-failover takes place based on a predefined condition of a monitored IP, when the condition is reversed, the IP becomes reachable on the original route, and the newly added route is deleted. Recovery is performed only when all RPM probes report the IP as reachable.

No Link Title lists the test parameters and its default values:

Table 80: Test Parameters and Default Values

Parameter	Default Value
probe-count	1
probe-interval	3 seconds
test-interval	1 second

No Link Title lists the supported threshold and its description:

Table 81: Threshold Supported and Description

Threshold	Description
Successive-Loss	Successive loss count of probes
Total-Loss	Total probe lost count

IP Monitoring Through Redundant Ethernet Interface Link Aggregation Groups

IP monitoring checks the reachability of an upstream device. It is designed to check the end-to-end connectivity of configured IP addresses and allows a redundancy group (RG) to automatically failover when the monitored IP address is not reachable through the redundant Ethernet. Both the primary and secondary devices in the chassis cluster monitor specific IP addresses to determine whether an upstream device in the network is reachable.

A redundant Ethernet interface contains physical interfaces from both the primary and secondary nodes in the SRX Series chassis cluster. In a redundant Ethernet interface, two physical interfaces are configured with each node contributing one physical interface. In a redundant Ethernet interface LAG, more than two physical interfaces are configured in the redundant Ethernet interface.

Example: Configure IP Monitoring on SRX Series Firewalls

IN THIS SECTION

- Requirements | 814
- Overview | 814
- Configuration | 814
- Verification | 817

This example shows how to monitor IP on an SRX Series Firewall.

Requirements

Before you begin:

Configure the following RPM options for RPM test:

- target-address
- probe-count
- probe-interval
- test-interval
- thresholds
- next-hop

Overview

This example shows how to set up IP monitoring on an SRX Series Firewall.

Configuration

IN THIS SECTION

- Procedure | 815

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, past them into a text file, remove any line breaks, change any details to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set services rpm probe Probe-Payment-Server test paysvr target address 1.1.1.10
set services rpm probe Probe-Payment-Server test paysvr probe-count 10
set services rpm probe Probe-Payment-Server test paysvr probe-interval 5
set services rpm probe Probe-Payment-Server test paysvr test-interval 5
set services rpm probe Probe-Payment-Server test paysvr thresholds successive-loss 10
set services rpm probe Probe-Payment-Server test paysvr next-hop 2.2.2.1
set services ip-monitoring policy Payment-Server-Tracking match rpm-probe Probe-Payment-Server
set services ip-monitoring policy Payment-Server-Tracking then preferred-route route 1.1.1.0/24
next-hop 1.1.1.99
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure IP monitoring on an SRX Series Firewall:

1. Configure the target address under the RPM probe.

```
[edit ]
user@host# set services rpm probe Probe-Payment-Server test paysvr target address 1.1.1.10
```

2. Configure the probe count under the RPM probe.

```
[edit ]
user@host# set services rpm probe Probe-Payment-Server test paysvr probe-count 10
```

3. Configure the probe interval (in seconds) under the RPM probe.

```
[edit ]
user@host# set services rpm probe Probe-Payment-Server test paysvr probe-interval 5
```

4. Configure the test interval (in seconds) under the RPM probe.

```
[edit ]
user@host# set services rpm probe Probe-Payment-Server test paysvr test-interval 5
```

5. Configure the threshold successive loss count under the RPM

```
[edit ]
user@host# set services rpm probe Probe-Payment-Server test paysvr thresholds successive-
loss 10
```

6. Configure the next-hop IP address under the RPM probe.

```
[edit ]
user@host# set services rpm probe Probe-Payment-Server test paysvr next-hop 2.2.2.1
```

7. Configure the IP monitoring policy under services.

```
[edit ]
user@host# set services ip-monitoring policy Payment-Server-Tracking match rpm-probe Probe-
Payment-Server
```

NOTE: The following steps are not mandatory. You can configure interface actions and route actions independently, or you can configure both the interface action and the route action together in one IP monitoring policy.

8. Configure the IP monitoring preferred route under services.

```
[edit ]
user@host# set services ip-monitoring policy Payment-Server-Tracking then preferred-route
route 1.1.1.0/24 preferred-metric 4
```

9. Configure the IP monitoring interface actions.

   - Enable

```
[edit ]
user@host# set services ip-monitoring policy Payment-Server-Tracking then interface
ge-0/0/1 enable
```

   - Disable

```
[edit ]
user@host# set services ip-monitoring policy Payment-Server-Tracking then interface
fe-0/0/[4-6] disable
```

10. Configure the no-preempt option.

```
[edit ]
user@host# set services ip-monitoring policy Payment-Server-Tracking no-preempt
```

Verification

IN THIS SECTION

- Verifying IP Monitoring | 817

Verifying IP Monitoring

Purpose

Verify the IP monitoring status of a policy.

Action

To verify the configuration is working properly, enter the following command:

```
show services ip-monitoring status <policy-name>
```

Example: Configure IP Monitoring on SRX5000 line

IN THIS SECTION

- Requirements | 818
- Overview | 818
- Configuration | 820
- Verification | 823

This example shows how to monitor SRX Series Firewalls with chassis cluster enabled.

Requirements

- You need two SRX5800 Services Gateways with identical hardware configurations, one SRX Series Firewall and one EX8208 Ethernet Switch.
- Physically connect the two SRX5800 devices (back-to-back for the fabric and control ports) and ensure that they are the same models. Configure/add these two devices in a cluster.

Overview

IN THIS SECTION

- Topology | 819

IP address monitoring checks end-to-end reachability of configured IP address and allows a redundancy group to automatically fail over when not reachable through the child link of redundant Ethernet interface (known as a reth) interface. Redundancy groups on both devices in a cluster can be configured to monitor specific IP addresses to determine whether an upstream device in the network is reachable.

When you configure multiple IP addresses on the reth Interface in a chassis cluster setup, IP monitoring uses the first IP address from the list of IP addresses configured for that reth interface on the primary node, and the first IP address from the list of secondary IP addresses configured for that reth interface on the backup node. The first IP address is the one with smallest prefix (netmask).

This example shows how to set up IP monitoring on an SRX Series Firewall.

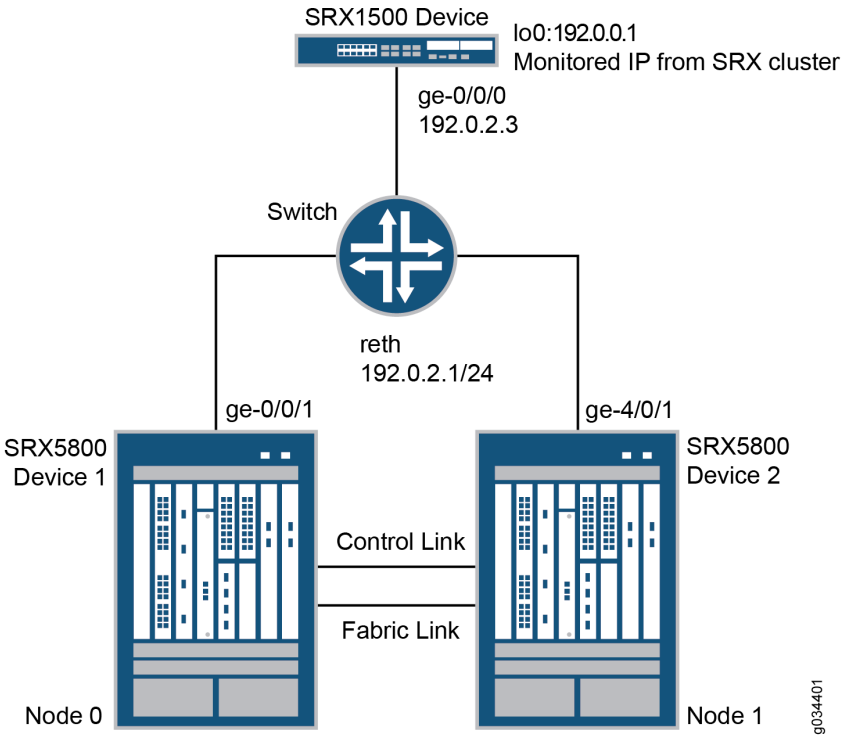
NOTE: IP monitoring is not supported on an NP-IOC card.

NOTE: IP monitoring does not support MIC online/offline status on SRX Series Firewalls.

Topology

Figure 30 on page 819 shows the topology used in this example.

Figure 30: IP Monitoring on an SRX Series Firewall Topology Example



In this example, two SRX5800 devices in a chassis cluster are connected to an SRX1500 device through an EX8208 Ethernet Switch. The example shows how the redundancy groups can be configured to

monitor key upstream resources reachable through redundant Ethernet interfaces on either node in a cluster.

Configuration

IN THIS SECTION

- CLI Quick Configuration | **820**
- Configuring IP Monitoring on SRX Series Firewall | **821**

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set chassis cluster reth-count 1
set chassis cluster redundancy-group 0 node 0 priority 254
set chassis cluster redundancy-group 0 node 1 priority 1
set chassis cluster redundancy-group 1 node 0 priority 200
set chassis cluster redundancy-group 1 node 1 priority 199
set chassis cluster redundancy-group 1 ip-monitoring global-weight 255
set chassis cluster redundancy-group 1 ip-monitoring global-threshold 80
set chassis cluster redundancy-group 1 ip-monitoring retry-interval 3
set chassis cluster redundancy-group 1 ip-monitoring retry-count 10
set chassis cluster redundancy-group 1 ip-monitoring family inet 192.0.0.1 weight 80
set chassis cluster redundancy-group 1 ip-monitoring family inet 192.0.0.1 interface reth0.0
secondary-ip-address 192.0.2.2
set interfaces ge-0/0/1 gigether-options redundant-parent reth0
set interfaces ge-4/0/1 gigether-options redundant-parent reth0
set interfaces reth0 redundant-ether-options redundancy-group 1
set interfaces reth0 unit 0 family inet address 192.0.2.1/24
set routing-options static route 192.0.0.1/32 next-hop 192.0.2.3
```

Configuring IP Monitoring on SRX Series Firewall

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure IP monitoring on an SRX Series Firewall:

1. Specify the number of redundant Ethernet interfaces.

```
{primary:node0}[edit]
user@host# set chassis cluster reth-count 1
```

2. Specify a redundancy group's priority for primacy on each node of the cluster. The higher number takes precedence.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 0 node 0 priority 254
user@host# set chassis cluster redundancy-group 0 node 1 priority 1
user@host# set chassis cluster redundancy-group 1 node 0 priority 200
user@host# set chassis cluster redundancy-group 1 node 1 priority 199
```

3. Configure the redundant Ethernet interfaces to redundancy-group 1.

```
{primary:node0}[edit]
user@host# set interfaces reth0 redundant-ether-options redundancy-group 1
user@host# set interfaces reth0 unit 0 family inet address 192.0.2.1/24
```

4. Assign child interfaces for the redundant Ethernet interfaces from node 0 and node 1.

```
{primary:node0}[edit]
user@host# set interfaces ge-0/0/1 gigether-options redundant-parent reth0
user@host# set interfaces ge-4/0/1 gigether-options redundant-parent reth0
```

5. Configure the static route to the IP address that is to be monitored.

```
{primary:node0}[edit]
user@host# set routing-options static route 192.0.0.1/32 next-hop 192.0.2.3
```

6. Configure IP monitoring under redundancy-group 1 with global weight and global threshold.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring global-weight 255
user@host# set chassis cluster redundancy-group 1 ip-monitoring global-threshold 80
```

7. Specify the retry interval.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring retry-interval 3
```

8. Specify the retry count.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring retry-count 10
```

9. Assign a weight to the IP address to be monitored, and configure a secondary IP address that will be used to send ICMP packets from the secondary node to track the IP being monitored.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring family inet 192.0.0.1 weight
80
user@host# set chassis cluster redundancy-group 1 ip-monitoring family inet 192.0.0.1
interface reth0.0 secondary-ip-address 192.0.2.2
```

NOTE:

- The redundant Ethernet (reth0) IP address, **192.0.2.1/24**, is used to send ICMP packets from node 0 to check the reachability of the monitored IP.

- The secondary IP address, **192.0.2.2**, should belong to the same network as the reth0 IP address.
- The secondary IP address is used to send ICMP packets from node 1 to check the reachability of the monitored IP.

Verification

IN THIS SECTION

- Verifying Chassis Cluster Status— Before Failover | **823**
- Verifying Chassis Cluster IP Monitoring Status— Before Failover | **824**
- Verifying Chassis Cluster Status— After Failover | **824**
- Verifying Chassis Cluster IP Monitoring Status— After Failover | **825**

Confirm the configuration is working properly.

Verifying Chassis Cluster Status— Before Failover

Purpose

Verify the chassis cluster status, failover status, and redundancy group information before failover.

Action

From operational mode, enter the `show chassis cluster status` command.

```
show chassis cluster status

Cluster ID: 11
Node Priority Status Preempt Manual failover
Redundancy group: 0 , Failover count: 0
node0 254 primary no no
node1 1 secondary no no
Redundancy group: 1 , Failover count: 0
```

```
node0 200 primary no no
node1 199 secondary no no
```

Verifying Chassis Cluster IP Monitoring Status— Before Failover

Purpose

Verify the IP status being monitored from both nodes and the failover count for both nodes before failover.

Action

From operational mode, enter the `show chassis cluster ip-monitoring status redundancy-group 1` command.

```
show chassis cluster ip-monitoring status redundancy-group 1

node0:
--------------------------------------------------------------------------
Redundancy group: 1
IP address Status Failure count Reason
192.0.0.1 reachable 0 n/a
node1:
--------------------------------------------------------------------------
Redundancy group: 1
IP address Status Failure count Reason
192.0.0.1 reachable 0 n/a
```

Verifying Chassis Cluster Status— After Failover

Purpose

Verify the chassis cluster status, failover status, and redundancy group information after failover.

NOTE: If the IP address is not reachable, the following output will be displayed.

Action

From operational mode, enter the `show chassis cluster status` command.

```
show chassis cluster status

Cluster ID: 11
Node Priority Status Preempt Manual failover
Redundancy group: 0 , Failover count: 0
node0 254 primary no no
node1 1 secondary no no
Redundancy group: 1 , Failover count: 1
node0 0 secondary no no
node1 199 primary no no
```

Verifying Chassis Cluster IP Monitoring Status— After Failover

Purpose

Verify the IP status being monitored from both nodes and the failover count for both nodes after failover.

Action

From operational mode, enter the `show chassis cluster ip-monitoring status redundancy-group 1` command.

```
show chassis cluster ip-monitoring status redundancy-group 1

node0:
--------------------------------------------------------------------------
Redundancy group: 1
IP address Status Failure count Reason
192.0.0.1 unreachable 1 unknown
node1:
--------------------------------------------------------------------------
Redundancy group: 1
IP address Status Failure count Reason
192.0.0.1 reachable 0 n/a
```

RELATED DOCUMENTATION

Example: Configuring an Active/Passive Chassis Cluster on SRX5800 Devices

Example: Configure Chassis Cluster Redundancy Group IP Address Monitoring

IN THIS SECTION

- Requirements | 826
- Overview | 826
- Configuration | 827
- Verification | 829

This example shows how to configure redundancy group IP address monitoring for an SRX Series Firewall in a chassis cluster.

Requirements

Before you begin:

- Set the chassis cluster node ID and cluster ID. See *Example: Setting the Node ID and Cluster ID for Security Devices in a Chassis Cluster*
- Configure the chassis cluster management interface. See *Example: Configuring the Chassis Cluster Management Interface*.
- Configure the chassis cluster fabric. See *Example: Configuring the Chassis Cluster Fabric Interfaces*.

Overview

You can configure redundancy groups to monitor upstream resources by pinging specific IP addresses that are reachable through redundant Ethernet interfaces on either node in a cluster. You can also configure global threshold, weight, retry interval, and retry count parameters for a redundancy group. When a monitored IP address becomes unreachable, the weight of that monitored IP address is deducted from the redundancy group IP address monitoring global threshold. When the global threshold reaches 0, the global weight is deducted from the redundancy group threshold. The retry interval determines the ping interval for each IP address monitored by the redundancy group. The pings are sent

as soon as the configuration is committed. The retry count sets the number of allowed consecutive ping failures for each IP address monitored by the redundancy group.

In this example, you configure the following settings for redundancy group 1:

- IP address to monitor—10.1.1.10
- IP address monitoring global-weight—100
- IP address monitoring global-threshold—200

  The threshold applies cumulatively to all IP addresses monitored by the redundancy group.
- IP address retry-interval—3 seconds
- IP address retry-count—10
- Weight—100
- Redundant Ethernet interface—reth1.0
- Secondary IP address—10.1.1.101

Configuration

IN THIS SECTION

- Procedure | 827

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
{primary:node0}[edit]
user@host#
set chassis cluster redundancy-group 1 ip-monitoring global-weight 100
set chassis cluster redundancy-group 1 ip-monitoring global-threshold 200
set chassis cluster redundancy-group 1 ip-monitoring retry-interval 3
```

```
set chassis cluster redundancy-group 1 ip-monitoring retry-count 10
set chassis cluster redundancy-group 1 ip-monitoring family inet 10.1.1.10 weight 100 interface
reth1.0 secondary-ip-address 10.1.1.101
```

Step-by-Step Procedure

To configure redundancy group IP address monitoring:

1. Specify a global monitoring weight.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring global-weight 100
```

2. Specify the global monitoring threshold.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring global-threshold 200
```

3. Specify the retry interval.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring retry-interval 3
```

4. Specify the retry count.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring retry-count 10
```

5. Specify the IP address to be monitored, weight, redundant Ethernet interface, and secondary IP address.

```
{primary:node0}[edit]
user@host# set chassis cluster redundancy-group 1 ip-monitoring family inet 10.1.1.10 weight
100 interface reth1.0 secondary-ip-address 10.1.1.101
```

Results

From configuration mode, confirm your configuration by entering the `show chassis cluster redundancy-group 1` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

For brevity, this `show` command output includes only the configuration that is relevant to this example. Any other configuration on the system has been replaced with ellipses (...).

```
{primary:node0}[edit]
user@host# show chassis cluster redundancy-group 1
ip-monitoring {
    global-weight 100;
    global-threshold 200;
    family {
        inet {
            10.1.1.10 {
                weight 100;
                interface reth1.0 secondary-ip-address 10.1.1.101;
            }
        }
    }
}
```

If you are done configuring the device, enter `commit` from configuration mode.

Verification

IN THIS SECTION

- Verifying the Status of Monitored IP Addresses for a Redundancy Group | **829**

Verifying the Status of Monitored IP Addresses for a Redundancy Group

Purpose

Verify the status of monitored IP addresses for a redundancy group.

Action

From operational mode, enter the `show chassis cluster ip-monitoring status` command. For information about a specific group, enter the `show chassis cluster ip-monitoring status redundancy-group` command.

```
{primary:node0}
user@host> show chassis cluster ip-monitoring status
node0:
--------------------------------------------------------------------------

Redundancy group: 1
Global threshold: 200
Current threshold: -120

IP address                Status        Failure count  Reason    Weight
10.1.1.10                 reachable       0            n/a       100
10.1.1.101                reachable       0            n/a       100

node1:
--------------------------------------------------------------------------

Redundancy group: 1
Global threshold: 200
Current threshold: -120

IP address                Status        Failure count  Reason    Weight
10.1.1.10                 reachable       0            n/a       100
10.1.1.101                reachable       0            n/a       100
```

CHAPTER 9

sFlow Monitoring Technology

IN THIS CHAPTER

- sFlow Technology Overview | 831
- sFlow Support on Switches | 832
- Example: Configure sFlow for EVPN-VXLAN Networks | 839
- sFlow Support on Routers | 845
- Example: Configure sFlow Technology to Monitor Network Traffic | 849
- sFlow Agent Address Assignment | 857

sFlow Technology Overview

IN THIS SECTION

- Benefits of sFlow Technology | 832

The sFlow technology is a monitoring technology for high-speed switched or routed networks. sFlow monitoring technology collects samples of network packets and sends them in a UDP datagram to a monitoring station called a *collector*. You can configure sFlow technology on a device to monitor traffic continuously at wire speed on all interfaces simultaneously. You must enable sFlow monitoring on each interface individually; you cannot globally enable sFlow monitoring on all interfaces with a single configuration statement. Junos OS supports the sFlow technology standard described in RFC 3176, *InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks* (see http://faqs.org/rfcs/rfc3176.html).

sFlow technology implements the following two sampling mechanisms:

- Packet-based sampling—Samples one packet out of a specified number of packets from an interface enabled for sFlow technology. Only the first 128 bytes of each packet are sent to the collector. Data collected include the Ethernet, IP, and transport layer headers, along with other application-level

headers (if present). Although this type of sampling might not capture infrequent packet flows, the majority of flows are reported over time, allowing the collector to generate a reasonably accurate representation of network activity. You configure packet-based sampling when you specify a sample rate.

- Time-based sampling—Samples interface statistics (counters) at a specified interval from an interface enabled for sFlow technology. Statistics such as Ethernet interface errors are captured. You configure time-based sampling when you specify a polling interval.

  Interface statistics are the source of time-based sampling. Time-based sampling provides statistical data in the output of the `show interface statistics` command. If you clear the interface statistics using the command `clear interfaces statistics`, time-based sampling displays the reset values.

Benefits of sFlow Technology

- sFlow can be used by software tools like a network analyzer to continuously monitor tens of thousands of switch or router ports simultaneously.
- Because sFlow uses network sampling (forwarding one packet from *n* number of total packets) for analysis, it is not resource intensive (for example processing, memory and more). The sampling is done at the hardware application-specific integrated circuits (ASICs) and, hence, it is simple and more accurate.

sFlow Support on Switches

IN THIS SECTION

- sFlow for IP-over-IP Tunnels | **834**
- sFLow for QFabric System | **834**
- sFlow for EVPN-VXLAN | **835**
- sFlow Limitations on Switches | **838**

sFlow technology on the switches samples only raw packet headers. A raw Ethernet packet is the complete Layer 2 network frame.

An sFlow monitoring system consists of an sFlow agent embedded in the device (switch) and up to four external collectors. The sFlow agent's two main activities are random sampling and statistics gathering. The sFlow agent performs packet sampling and gathers interface statistics, and then combines the

information into UDP datagrams that are sent to the sFlow collectors. An sFlow collector can be connected to the switch through the management network or data network. The software forwarding infrastructure daemon (SFID) on the switch looks up the next-hop address for the specified collector IP address to determine whether the collector is reachable by way of the management network or data network.

Each datagram contains the following information:

- The IP address of the sFlow agent
- The number of samples
- The interface through which the packets entered the agent
- The interface through which the packets exited the agent
- The source and destination interface for the packets
- The source and destination VLAN for the packets

You can view the **Extended router data** and **Extended switch data** headers on collector as part of sFlow records.

The **Extended switch data** contains information of *Flow data length (byte), Incoming 802.1Q VLAN, Incoming 802.1p priority, Outgoing 802.1Q VLAN, and Outgoing 802.1p priority* fields

The **Extended router data** contains information of *Flow data length (byte), Next hop, Next hop source mask, and Next hop destination mask* fields.

EX Series switches adopt the distributed sFlow architecture. The sFlow agent has two separate sampling entities that are associated with each Packet Forwarding Engine. These sampling entities are known as subagents. Each subagent has a unique ID that is used by the collector to identify the data source. A subagent has its own independent state and forwards its own sample packets to the sFlow agent. The sFlow agent is responsible for packaging the samples into datagrams and sending them to the sFlow collector. Because sampling is distributed across subagents, the protocol overhead associated with sFlow technology is significantly reduced at the collector.

For the EX9200 switch and MX Series routers, we recommend that you configure the same sample rate for all the ports in a line card. If you configure different sample rates, the lowest value is used for all ports on the line card.

In case of dual VLANs, all fields may not be reported.

If the primary-role assignment changes in a *Virtual Chassis* setup, sFlow technology continues to function.

sFlow for IP-over-IP Tunnels

Starting in Junos OS Release 20.4R1, you can use sFlow technology to sample IP-over-IP traffic at a physical port on QFX5100 and QFX5200 devices. This feature is supported for IP-over-IP tunnels with an IPv4 outer header that carry IPv4 or IPv6 traffic. Use sFlow monitoring technology to randomly sample network packets from IP-over-IP tunnels and send the samples to a destination collector for monitoring. Devices that act as a IP-over-IP tunnel entry point, transit device, or tunnel endpoint support sFlow sampling. No Link Title shows the fields that are reported when a packet is sampled at the ingress or egress interface of a device that acts as an IP-over-IP tunnel entry point, transit device, or tunnel endpoint.

Table 82: Supported Metadata

sFlow Field	Tunnel Entry Point	Transit Device	Tunnel Endpoint
Raw packet header	Includes payload only	Includes payload and tunnel header	Egress: Includes payload only Ingress: Includes payload and tunnel header
Input interface	Incoming IFD SNMP index	Incoming IFD SNMP index	Incoming IFD SNMP index
Output interface	Outgoing IFD SNMP index	Outgoing IFD SNMP index	Outgoing IFD SNMP index

sFLow for QFabric System

On a QFabric system, sFlow technology monitors the interfaces on each Node device as a group, and implements the binary backoff algorithm based on the traffic on that group of interfaces.

On the QFabric system, the following default values are used if the optional parameters are not configured:

- Agent ID is the management IP address of the default partition.
- Source IP is the management IP address of the default partition.

In addition, the QFabric system subagent ID (which is included in the sFlow datagrams) is the ID of the Node group from which the datagram is sent to the collector.

On a QFabric system, the sFlow technology architecture is distributed. The global sFlow technology configuration defined on the QFabric system Director device is distributed to Node groups that have sFlow sampling configured on their interfaces. The sFlow agent has a separate sampling entity, known as

a *subagent*, running on each Node device. Each subagent has its own independent state and forwards its own sample information (datagrams) directly to the sFlow collectors.

On the QFabric system, an sFlow collector must be reachable through the data network. Because each Node device has all routes stored in the default routing instance, the collector IP address should be included in the default routing instance to ensure the collector's reachability from the Node device.

Regardless of the rate of traffic or the configured sampling interval, a datagram is sent whenever its size reaches the maximum Ethernet transmission unit (MTU) of 1500 bytes, or whenever a 250-ms timer expires, whichever occurs first. The timer ensures that a collector receives regularly sampled data.

Packet-based sampling in sFlow is implemented in the hardware. If traffic levels are unusually high, the hardware generates more samples than it can handle, and the extra samples are dropped, producing inaccurate results. Enabling the `disable-sw-rate-limiter` statement disables the software rate-limiting algorithm and allows the hardware sampling rate to stay within the maximum sampling rate.

sFlow for EVPN-VXLAN

On QFX10000 Series switches you can use sFlow technology to sample known multicast traffic carried over EVPN-VXLAN. Sampling of known multicast traffic is supported for traffic that enters the switch over EVPN-VXLAN or in other words core facing interface and egresses the switch out of customer-facing ports. Also, known multicast traffic sampling is supported only in the egress direction. To enable egress sFlow sampling of known multicast traffic on a customer facing port, you need to enable sFlow on the interface in the egress direction as it is done for the standard unicast traffic sampling scenario. In addition, you need to include the `egress-multicast enable` option at the `[edit forwarding options sflow]` hierarchy level. The maximum replication rate for multicast traffic samples can be configured using the `eggress-multicast max-replication-rate` *rate* option at the `[edit forwarding options sflow eggress-multicast]` hierarchy level.

When a set of sFlow egress sampling enabled interfaces are subscribed to a given multicast group and egress sFlow multicast sampling option is enabled, all the interfaces will be sampled at the same rate. The minimum of the configured sFlow rate, or in other words, the most aggressive sampling rate among this set of interfaces is used for sampling across all the interfaces in the set. A single port will generate samples at different rates if it is part of multiple multicast groups, as multicast sampling for a specific group depends on the most aggressive sampling rate among the ports of that particular group.

On EVPN-VXLAN, the centrally-routed bridging (CRB) and Edge-routed bridging (ERB) architecture are supported with sFlow. EVPN-VXLAN supports only IPv4 address.

Table 83: Supported Metadata

Incoming Interface and Encapsulation	Outgoing Interface and Encapsulation	Required Sampled Content	Forwarding Scenario	Metadata
Access port Layer 2 traffic	Network port	Incoming Layer 2 header + Layer 2 payload	Packets are encapsulated with VXLAN header and forwarded.	Incoming Interface Index or Identifier. Outgoing Interface Index or Identifier
Network port Layer 3 traffic	Access port	Incoming Layer 3 header + VXLAN header + Inner payload	Packets are de-capsulated and forwarded.	Incoming Virtual Tunnel End Point (VTEP) Interface Index or Identifier. Outgoing Interface Index or Identifier
Access port Layer 2 traffic	Network port	Incoming Layer 2 Header + Layer 2 payload	Packets are encapsulated with VXLAN header and forwarded.	Incoming Interface Index or Identifier. Outgoing Interface Index or Identifier
Network port Layer 3 traffic	Access port	Inner payload	Packets are de-capsulated and forwarded.	Incoming VTEP Interface Index or Identifier. Outgoing Interface Index or Identifier

No Link Title provides Metadata information for extended switch data and extended routing data.

Table 84: Supported Metadata for Extended Switch Data and Extended Routing Data

EVPN-VXLAN	Scenario	Traffic Type	sFlow Interface Side	VXLAN Tunnel Type	Extended Switch Data				Extended Routing Data		
					IIF VLAN	IIF VLAN Priority	OIF VLAN	OIF VLAN Priority	NH IP	NH SMASK	NH DMASK
CRB	Layer 2 GW Leaf	Layer 2	Ingress	Encap	Yes	Yes	No	No	Yes	Yes	Yes
				Decap	No	No	Yes	No	No	No	No
			Egress	Encap	Yes	No	No	No	Yes	Yes	Yes
				Decap	No	No	Yes	No	No	No	No
	Layer 3 GW Spine	Layer 2	Ingress	No	No	No	No	No	No	No	No
				No	No	No	No	No	No	No	No
				Transit	No	No	No	No	Yes	Yes	Yes
			Egress	No	No	No	No	No	No	No	No
				No	No	No	No	No	No	No	No
				Transit	No	No	No	No	Yes	Yes	Yes
		Layer 3 Traffic (Inter Vlan Case)	Ingress	Encap	No	No	No	No	Yes	Yes	Yes
				Decap	No	No	No	No	Yes	Yes	Yes
				Transit	No	No	No	No	Yes	Yes	Yes
			Egress	Encap	No	No	No	No	Yes	Yes	Yes
				Decap	No	No	No	No	Yes	Yes	Yes

Table 84: Supported Metadata for Extended Switch Data and Extended Routing Data *(Continued)*

EVPN-VXLAN	Scenario	Traffic Type	sFlow Interface Side	VXLAN Tunnel Type	Extended Switch Data				Extended Routing Data		
					IIF VLAN	IIF VLAN Priority	OIF VLAN	OIF VLAN Priority	NH IP	NH SMASK	NH DMASK
				Transit	No	No	No	No	Yes	Yes	Yes
ERB	Layer 2+Layer 3	Layer 2	Ingress	Encap	Yes	Yes	No	No	Yes	Yes	Yes
				Decap	No	No	Yes	No	No	No	No
			Egress	Encap	Yes	No	No	No	Yes	No	Yes
				Decap	No	No	Yes	No	No	No	No
		Layer 3 Traffic (Inter VLAN Case)	Ingress	Encap	Yes	Yes	No	No	Yes	Yes	Yes
				Decap	No	No	Yes	No	No	No	No
			Egress	Encap	Yes	No	No	No	Yes	Yes	Yes
				Decap	No	No	Yes	No	No	No	No

sFlow Limitations on Switches

On switches, limitations of sFlow traffic sampling include the following:

- The EX3400, EX4100, EX4300, EX4400, and QFX5K series switches use pseudo-egress sampling for egress sampling. Packets are not true egress samples. They are unmodified copies as they appear in the ingress pipeline of the sflow instance device that is using egress sampling.
- On QFX5130-32CD and QFX5700 devices, the egress sFlow uses the ingress pipeline packet, unlike other QFX series devices that use original source and destination IP addresses. The sampled packets at the egress interface show the VXLAN header with the ingress VXLAN's source and destination IP addresses.

The egress sampled packets for the QFX5130-32CD and QFX5700 devices show the IP addresses of the VXLAN endpoints from the preceding VXLAN tunnel. The command `show interfaces vtep extensive` displays that the sampled packets are routed through the VXLAN VTEP interface. This is not true egress sampling.

- On EX9200 switches, true OIF (outgoing interface) is not supported with sFlow.

EX9200 switches support configuration of only one sampling rate (inclusive of ingress and egress rates) on an FPC (or line card). To support compatibility with the sFlow configuration of other Juniper Networks products, EX9200 switches still accept multiple rate configuration on different interfaces of the same FPC. However, the switch programs the lowest rate as the sampling rate for all the interfaces of that FPC. The (`show sflow interfaces`) command displays the configured rate and the actual (effective) rate. However, different rates on different FPCs is still supported on EX9200 switches.

Example: Configure sFlow for EVPN-VXLAN Networks

IN THIS SECTION

- Requirements | **839**
- Overview and Topology | **840**
- Configuration | **841**
- Verification | **843**

Use this example to configure and use sFlow monitoring for EVPN-VXLAN traffic with an IPv4 underlay on QFX10000 line of switches.

Requirements

This example uses the following hardware and software components:

- A QFX10002-60C, QFX10002, QFX10008, or QFX10016 switch.
- Junos OS Release 21.3R1, 21.2R2 and later.

This example assumes that you already have an EVPN-VXLAN with an IPv4 underlay based network and want to enable sFlow monitoring on a QFX10000 switch.

Overview and Topology

IN THIS SECTION

- Topology | 840

In this example, you enable sFlow inspection for an existing and working EVPN-VXLAN network traffic with IPv4 underlay.

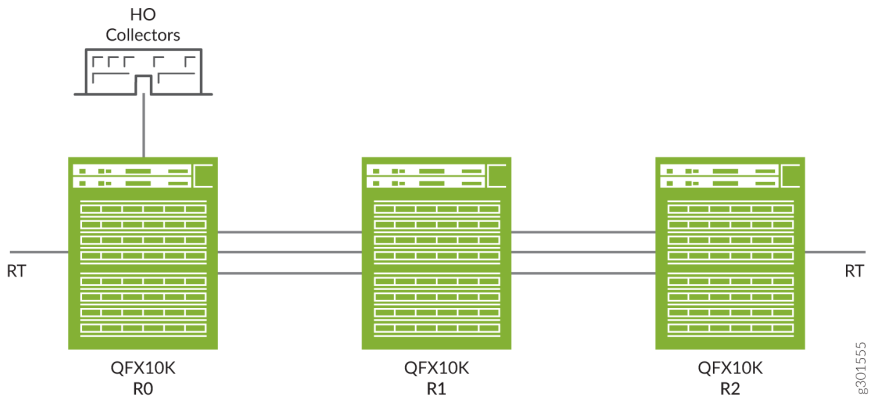
Topology

Figure 31 on page 841 depicts the sFlow support in an EVPN-VXLAN network environment with an IPv4 underlay. In this topology, the sFlow agent performs packet sampling and gathers interface statistics, and then combines the information into UDP datagrams that are sent to sFlow collectors. You can connect an sFlow collector to the switch through the management network or data network. The sFlow program on the switch looks up the next-hop address for the specified collector IP address to determine whether the collector is reachable by way of the management network or data network.

You should configure sFlow on the physical port of your hardware switch and logical interface where the VTEPs (virtual port) are configured and not on VTEPs itself. When you configure sFlow on fabric facing interface, the underlay traffic along with VXLAN traffic is sampled. You can configure sFlow on any of the R0, R1, or R2 devices mentioned in the topology.

For information about basic EVPN-VXLAN underaly configuration, refer to Example: Configuring a QFX10000 Switch as a Layer 3 VXLAN Gateway in an EVPN-VXLAN Centrally-Routed Bridging Overlay.

Figure 31: sFlow Support on EVPN-VXLAN Network



Configuration

IN THIS SECTION

- CLI Quick Configuration | 841
- Step-by-Step Procedure | 842
- Results | 843

Use the following steps to configure sFlow technology on your QFX10000 switch with EVPN-VXLAN network:

CLI Quick Configuration

To quickly configure this example on your QFX10000 switch, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the [edit] hierarchy level.

```
[edit protocols sflow]
set polling-interval 20
set sample-rate ingress 10
set source-ip 10.1.12.0
set collector 10.102.70.200set interfaces et-0/0/1.1 sample-rate ingress 100 egress 100
```

Step-by-Step Procedure

To configure sFlow technology:

1. Specify in seconds how often the sFlow agent polls the interface:

```
[edit protocols sflow]
user@switch# set polling-interval 0
```

2. Specify the rate at which ingress packets must be sampled:

```
[edit protocols sflow]
user@switch# set sample-rate ingress 100
```

3. Configure the source IP address:

```
[edit protocols sflow]
user@switch# set source-ip 10.1.12.0
```

4. Configure the IP address of the collector:

```
[edit protocols sflow]
user@switch# set collector 192.168.200.100
```

5. Enable sFlow technology on a specific interface:

```
[edit protocols sflow]
user@switch# set interfaces et-0/0/1.1 sample rate ingress 100 egress 100
```

6. Commit the configuration:

```
[edit protocols sflow]
user@switch# commit
```

Results

Check the results of the configuration:

```
[edit]
user@switch#  show protocols sflow
agent-id 10.1.12.0/24;
polling-interval 0;
sample-rate {
ingress 16000;
egress 16000;
}
collector 192.168.200.100;
interfaces et-0/0/54.1 {
sample-rate {
ingress 100;
egress 100;
}
}
interfaces et-0/0/56.0;
interfaces et-0/0/57.1 {
sample-rate {
ingress 100;
egress 100;
}
}
```

Verification

IN THIS SECTION

- Verify Configured sFlow Technology | **844**

To confirm that the sFlow configuration is enabled and correct.

Verify Configured sFlow Technology

IN THIS SECTION

- Purpose | 844
- Action | 844

Purpose

Verify the sFlow monitoring is enabled for an EVPN-VXLAN network.

Action

From operational mode, enter the `show protocols sflow` command.

```
user@switch> show protocols sflow
sFlow                    : Enabled
Adaptive fallback        : Disabled
Sample limit             : 300 packets/second
Sample limit Threshold   : 0 packets/second
Polling interval         : 0 second
Sample rate egress       : 1:2048: Disabled
Sample rate ingress      : 1:100: Enabled
Agent ID                 : 10.1.12.0/24
Source IP address        : 10.1.12.0
```

RELATED DOCUMENTATION

Understanding Flexible Ethernet Services Support With EVPN-VXLAN

No Link Title

sFlow Support on Routers

IN THIS SECTION

- sFlow for GRE Encapsulation | 846
- sFlow Sample Size | 848
- sFlow Limitations on Routers | 849

On PTX1000 routers and QFX10000 Series switches, sFlow technology always works at the level of the physical interface. Enabling sFlow monitoring on one logical interface enables it on all logical interfaces belonging to that physical interface.

On PTX1000 routers, PTX10000 routers, and QFX10000 Series switches, you can configure sFlow only on an active logical interface. Use the `show interfaces terse` command to display the status information of interfaces. If both operational and admin state of an interface is up, then it is an active interface.

On PTX10000 routers, PTX5000 routers and QFX10000 Series switches, sFlow will not generate samples as expected when the ingress or egress interfaces are part of routing instance specifically in ECMP scenario.

The sFlow agent is responsible for monitoring the network port, sample all incoming packets including control traffic and traffic arriving on all the ports in the system.

sFlow technology is supported only on the ACX5000 line of routers, other ACX Series routers do not support this technology.

The following sFlow features are supported on the ACX5000 line of routers:

- Packet-based sampling

  NOTE: This feature is not supported on ACX5448 router.

- Time-based sampling
- Adaptive sampling

The following sFlow technology limitations apply on ACX5000 line of routers:

- The ingress and egress sampling can be configured only on one of the units under a physical interface and the sFlow is enabled for the physical interface (port). The sFlow cannot be enabled if the unit under a physical interface is not configured.
- Egress sampling for Broadcast, Unknown unicast and Multicast (BUM) traffic is not supported because the **source-interface** field in the sFlow datagrams cannot be populated.
- Destination VLAN and Destination Priority fields are not populated in the case of Layer 3 forwarding.
- sFlow sampling is not supported on the output interface of an analyzer.
- SNMP MIB support for sFlow is not available.
- sFlow cannot be enabled on IRB interfaces.
- sFlow cannot be enabled on logical tunnel (lt-) and LSI interfaces.

sFlow for GRE Encapsulation

On PTX10001-36MR, PTX10003, PTX10004, PTX10008, and PTX10016 devices, sFlow supports the export of Extended Tunnel Egress Structure fields for traffic entering IPv4 or IPv6 GRE tunnels. This enables sFlow to provide information about GRE tunnel into which a packet entering the device might be encapsulated. The GRE tunnel could be IPv4 or IPv6. The feature is supported only when sFlow is enabled in the ingress direction wherein firewall based GRE encapsulation happens on IPv4 or IPv6 packets.

The feature is supported for the below traffic scenarios when ingress sFlow sampling is enabled:

- Incoming IPv4 traffic that undergoes IPv4 GRE encapsulation
- Incoming IPv6 traffic that undergoes IPv4 GRE encapsulation
- Incoming IPv4 traffic that undergoes IPv6 GRE encapsulation
- Incoming IPv6 traffic that undergoes IPv6 GRE encapsulation

To learn more about the sFlow and sFlow Tunnel Structures, see sFlow Tunnel Structures.

No Link Title describes extended tunnel egress structure fields for traffic entering IPv4 or IPv6 GRE tunnels.

Table 85: Extended Tunnel Egress Structure Fields and Values

Field Name	Value
Protocol reported	0x2f (GRE)

Table 85: Extended Tunnel Egress Structure Fields and Values *(Continued)*

Field Name	Value
Source IP	IPv4 or IPv6 address of the tunnel source
Destination IP	IPv4 or IPv6 address of the tunnel destination endpoint
length	0
source port	0
destination port	0
tcp flags	0
priority	0

The extended structure for IPv4 and IPv6 GRE tunnels is below:

```
/* opaque = flow_data; enterprise = 0; format = 1023 */

struct extended_ipv4_tunnel_egress {

    sampled_ipv4 header;

}
/* opaque = flow_data; enterprise = 0; format = 1025 */

struct extended_ipv6_tunnel_egress {

    sampled_ipv6 header;

}
```

Sampled IPv4 header structure is below:

```
/* Packet IP version 4 data */
/* opaque = flow_data; enterprise = 0; format = 3 */
```

```
struct sampled_ipv4 {
   unsigned int length;     /* The length of the IP packet excluding
                               lower layer encapsulations */
   unsigned int protocol;   /* IP Protocol type
                                  (for example, TCP = 6, UDP = 17) */
   ip_v4 src_ip;            /* Source IP Address */
   ip_v4 dst_ip;            /* Destination IP Address */
   unsigned int src_port;   /* TCP/UDP source port number or equivalent */
   unsigned int dst_port;   /* TCP/UDP destination port number or equivalent
   unsigned int tcp_flags;  /* TCP flags */
   unsigned int tos;        /* IP type of service */
}
```

Sampled IPv6 header structure is below:

```
/* Packet IP Version 6 Data */
/* opaque = flow_data; enterprise = 0; format = 4 */
struct sampled_ipv6 {
   unsigned int length;     /* The length of the IP packet excluding
                               lower layer encapsulations */
   unsigned int protocol;   /* IP next header
                               (for example, TCP = 6, UDP = 17) */
   ip_v6 src_ip;            /* Source IP Address */
   ip_v6 dst_ip;            /* Destination IP Address */
   unsigned int src_port;   /* TCP/UDP source port number or equivalent */
   unsigned int dst_port;   /* TCP/UDP destination port number or equivalent*/
   unsigned int tcp_flags;  /* TCP flags */
   unsigned int priority;   /* IP priority */
}
```

sFlow Sample Size

Starting in Junos OS Evolved 23.1R1 release for PTX Series devices, you can configure the sFlow sample size of the raw packet header to be exported as part of the sFlow record to the collector. The configurable range of sample size is from 128 bytes through 512 bytes. Use the `set protocols sflow sample-size` *`Sample-Size`* command to configure the sample size. If the configured sample size is greater than the actual packet size, then the actual size of the packet is exported. If you do not configure the sample size, the default size of the raw packet header exported to the collector is 128 bytes.

The sample size configured in the global sFlow configuration is inherited by all the interfaces configured under sFlow protocols.

sFlow Limitations on Routers

On routers, limitations of sFlow traffic sampling include the following:

- Trio chipset cannot support different sampling rate for each family. Hence, only one sampling rate can be supported per line card.
- Adaptive load balancingsampling is applied per line card and not for per interface under the line card.

Routers support configuration of only one sampling rate (inclusive of ingress and egress rates) on an line card. To support compatibility with the sFlow configuration of other Juniper Networks products, the routers still accept multiple rate configuration on different interfaces of the same line card. However, the router programs the lowest rate as the sampling rate for all the interfaces of that line card. The (`show sflow interfaces`) command displays the configured rate and the actual (effective) rate. However, different rates on different line cards is still supported on Juniper Networks routers.

In Junos OS Evolved, you can configure sFlow only on Ethernet interfaces (`et-*`) for the following PTX Series devices:

- PTX10003-80C and PTX10003-160C
- PTX10008
- PTX10001-36MR
- PTX10004
- PTX10016

You cannot configure sFlow on loopback interfaces (`lo0`).

Example: Configure sFlow Technology to Monitor Network Traffic

IN THIS SECTION

- Requirements | **850**
- Topology | **850**
- Configuration | **851**

Verification | 854

This example describes how to configure and use sFlow technology to monitor network traffic.

Requirements

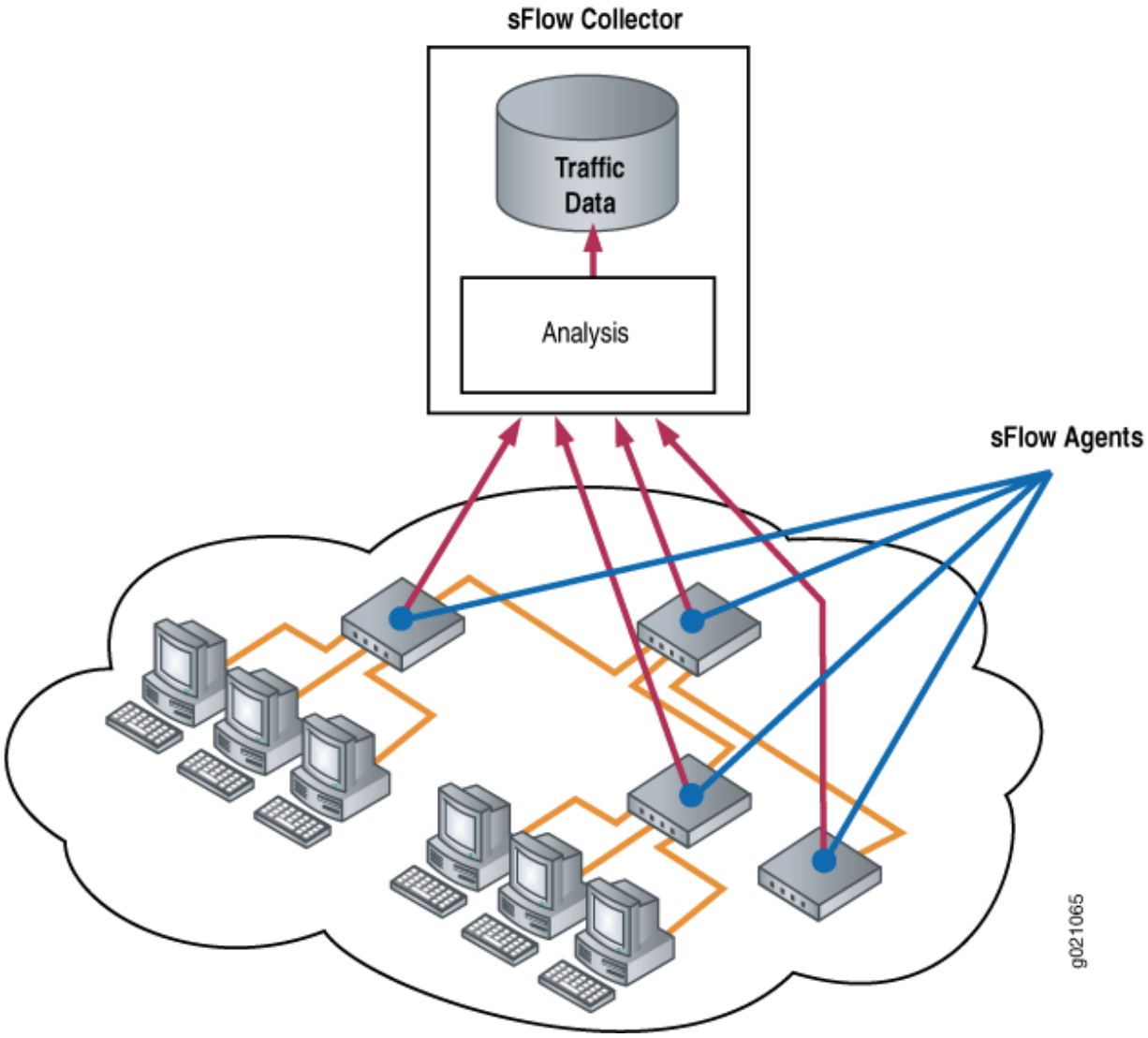
You can use QFX Series, EX Series, PTX Series and MX Series devices for the example using the following hardware and software components:

- One EX Series switch
- Junos OS Release 9.3 or later for EX Series switches
- One MX Series router
- Junos OS Release 18.1 or later for MX Series routers
- Junos OS Release 11.3 or later
- One QFX3500 switch

Topology

The sFlow agent runs on the switch. It combines interface counters and flow samples and sends them across the network to the sFlow collector. Figure 32 on page 851 depicts the basic elements of the sFlow system.

Figure 32: sFlow Technology Monitoring System



Configuration

IN THIS SECTION

- CLI Quick Configuration | **852**
- Procedure | **852**

To configure sFlow technology, perform the following tasks:

CLI Quick Configuration

To quickly configure sFlow technology, copy the following commands and paste them into the switch terminal window:

```
[edit protocols]
set sflow collector 10.204.32.46 udp-port 5600
set sflow interfaces ge-0/0/0
set sflow polling-interval 20
set sflow sample-rate egress 1000
```

Procedure

Step-by-Step Procedure

To configure sFlow technology:

1. Configure the IP address and UDP port of the collector:

```
[edit protocols]
user@switch# set sflow collector 10.204.32.46 udp-port 5600
```

NOTE: You can configure a maximum of 4 collectors.
The default UDP port is 6343.

2. Enable sFlow technology on a specific interface:

```
[edit protocols sflow]
user@switch# set interfaces ge-0/0/0
```

NOTE: You cannot enable sFlow technology on a Layer 3 VLAN-tagged interface.

3. Specify in seconds how often the sFlow agent polls the interface:

```
[edit protocols sflow]
user@switch# set polling-interval 20
```

> **NOTE**: The polling interval can be specified as a global parameter also. Specify **0** if you do not want to poll the interface.

4. Specify the rate at which egress packets must be sampled:

```
[edit protocols sflow]
user@switch# set sample-rate egress 1000
```

> **NOTE**: You can specify both egress and ingress sampling rates. If you set only the **egress** sampling rate, the **ingress** sampling rate will be disabled.

> **NOTE**: We recommend that you configure the same sampling rates on all the ports on a line card. If you configure different sampling rates are different, the lowest value is used for all ports. You could still configure different rates on different line cards.

5. (Optional) Specify the sample size for the raw packet header. The sample size configuration is applicable on PTX10003-80C, PTX10003-160C, PTX10001-36MR, PTX10004, PTX10008 and PTX10016 devices from 23.1R1 Junos OS Evolved release.

```
[edit protocols sflow]
user@switch# set sample-size 135
```

Results

Check the results of the configuration:

```
[edit protocols sflow]
user@switch# show
```

```
polling-interval 20;
    sample-rate egress 1000;
    collector 10.204.32.46 {
        udp-port 5600;
    }
interfaces ge-0/0/0.0;
```

```
[edit protocols sflow]
user@router# show
polling-interval 20;
source-ip 45.1.1.1;
collector 45.1.1.100;
sample-size 135;
```

Verification

IN THIS SECTION

- Verifying That sFlow Technology Is Configured Properly | **854**
- Verifying That sFlow Technology Is Enabled on the Specified Interface | **855**
- Verifying the sFlow Collector Configuration | **856**

To confirm that the configuration is correct, perform these tasks:

Verifying That sFlow Technology Is Configured Properly

Purpose

Verify that sFlow technology is configured properly.

Action

Use the `show sflow` command:

```
user@switch> show sflow
sFlow: Enabled
Sample limit: 300 packets/second
Polling interval: 20 seconds
Sample rate egress: 1:1000: Enabled
Sample rate ingress: 1:2048: Disabled
Agent ID: 10.204.96.222
```

```
user@router> show sflow
sFlow                    : Enabled
Adaptive fallback        : False
Sample limit             : 2000 packets/second
Sample limit Threshold   : 0 packets/second
Polling interval         : 20 second
Sample rate egress       : 1:2048:Disabled
Sample rate ingress      : 1:2048:Disabled
Agent ID                 : 10.204.96.222
Agent ID IPv6            : No valid agent IPv6
Source IP address        : 45.1.1.1
Source IPv6 address      : No valid source IPv6
Sample Size              : 128 Bytes
```

NOTE: The sampling limit cannot be configured and is set to 300 packets/second per FPC.

Meaning

The output shows that sFlow technology is enabled and specifies the values for the sampling limit, polling interval, and the egress sampling rate.

Verifying That sFlow Technology Is Enabled on the Specified Interface

Purpose

Verify that sFlow technology is enabled on the specified interfaces and display the sampling parameters.

Action

Use the `show sflow interface` command:

```
user@switch> show sflow interface
Interface        Status      Sample rate    Adapted sample rate   Polling-interval
            Egress Ingress   Egress Ingress   Egress Ingress
ge-0/0/0.0  Enabled  Disabled  1000    2048    1000     2048            20
```

Meaning

The output indicates that sFlow technology is enabled on the ge-0/0/0.0 interface with an egress sampling rate of 1000, a disabled ingress sampling rate, and a polling interval of 20 seconds.

Verifying the sFlow Collector Configuration

Purpose

Verify the sFlow collector's configuration.

Action

Use the `show sflow collector` command:

```
user@switch> show sflow collector

Collector         Udp-port    No. of samples
 address
10.204.32.46      5600        1000
10.204.32.76      3400        1000
```

```
user@router> show sflow collector
Collector                                Udp-port    Dscp     Forwarding-Class
No. of samples
 address
45.1.1.100                               6343        0        best-effort                    0
```

Meaning

The output displays the IP address of the collectors and the UDP ports. It also displays the number of samples.

sFlow Agent Address Assignment

The sFlow collector uses the sFlow agent's IP address to determine the source of the sFlow data. You can configure the IP address of the sFlow agent to ensure that the agent ID of the sFlow agent remains constant. If you do not specify the IP address to be assigned to the agent, an IP address is automatically assigned to the agent based on the following order of priority of interfaces configured on the device:

Table 86: Interfaces on the Devices

Routers and EX Series Switches	QFX Series Devices
1. Virtual Management Ethernet (VME) interface 2. Management Ethernet interface	1. Management Ethernet interface em0 IP address 2. Any Layer 3 interface if the em0 IP address is not available

If neither of the preceding interfaces has been configured, the IP address of any Layer 3 interface or the *routed VLAN interface* (*RVI*) is assigned to the agent. At least one interface must be configured on the switch for an IP address to be automatically assigned to the agent. When the agent's IP address is assigned automatically, the IP address is dynamic and changes when the switch reboots.

sFlow data can be used to provide network traffic visibility information. You can explicitly configure the IP address to be assigned to source data (sFlow datagrams). If you do not explicitly configure that address, the IP address of the configured Gigabit Ethernet interface, 10-Gigabit Ethernet interface, or the RVI is used as the source IP address.

CHAPTER 10

Adaptive Sampling for Routers and Switches

IN THIS CHAPTER

- Adaptive Sampling Overview | 858

Adaptive Sampling Overview

IN THIS SECTION

- How Adaptive Sampling Works | 860
- Adaptive Sampling Fallback | 860
- Adaptive Sampling Limitations | 861

Adaptive sampling is the process of monitoring the overall incoming traffic rate on the network device and providing intelligent feedback to interfaces to dynamically adapt the sampling rates on interfaces on the basis of traffic conditions. Adaptive sampling prevents the CPU from overloading and maintains the system at an optimum level, even when traffic patterns change on the interfaces. Whereas the *sample rate* is the configured number of egress or ingress packets out of which one packet is sampled, the *adaptive sample rate* is the maximum number of samples that should be generated per line card, that is, it's the limit given to adaptive sampling. *Sample load* is the amount of data (or number of packets) moving across a network at a given point of time that is sampled. As you increase the sample rate, you decrease the sample load and vice versa. For example, suppose the configured sample rate is 2 (meaning 1 packet out of 2 packets is sampled), and then that rate is doubled, making it 4, or only 1 packet out of 4 packets is sampled.

You configure the adaptive sample rate, which is the maximum number of samples that should be generated per line card, at the `[edit protocols sflow adaptive-sample-rate]` hierarchy level.

To ensure sampling accuracy and efficiency, QFX Series devices use adaptive sFlow sampling. Adaptive sampling monitors the overall incoming traffic rate on the device and provides feedback to the interfaces

to dynamically adapt their sampling rate to traffic conditions. The sFlow agent reads the statistics on the interfaces every 5 seconds and identifies five interfaces with the highest number of samples. On a standalone switch, when the CPU processing limit is reached, a binary backoff algorithm is implemented to reduce the sampling load of the top five interfaces by half. The adapted sampling rate is then applied to those top five interfaces.

Using adaptive sampling prevents overloading of the CPU and keeps the device operating at its optimum level even when there is a change in traffic patterns on the interfaces. The reduced sampling load is used until:

- You reboot the device.
- You configure a new sampling rate.
- The adaptive sampling fallback feature, if configured, increases the sampling load because the number of samples generated is less than the configured threshold.

If a particular interface is not configured, the IP address of the next interface in the priority list is used as the IP address for the agent. Once an IP address is assigned to the agent, the agent ID is not modified until the sFlow service is restarted. At least one interface has to be configured for an IP address to be assigned to the agent.

Considerations

On the QFX Series, limitations of sFlow traffic sampling include:

- sFlow sampling on ingress interfaces does not capture CPU-bound traffic.
- sFlow sampling on egress interfaces does not support broadcast and multicast packets.
- Egress samples do not contain modifications made to the packet in the egress pipeline.
- If a packet is discarded because of a firewall filter, the reason code for discarding the packet is not sent to the collector.
- The `out-priority` field for a VLAN is always set to 0 (zero) on ingress and egress samples.
- You cannot configure sFlow monitoring on a link aggregation group (LAG), but you can configure it individually on a LAG member interface.
- On QFX10000 Series switches, for a set of ports in a multicast group, since the actual sampling happens in the ingress pipeline for egress packets, the minimum of the configured sFlow rate or the most aggressive sample rate among those ports is used for sampling across all ports in that group.
- Starting from Junos OS Release 19.4 and later, on QFX10000 Series switches, if the destination port of a sampled UDP packet is 6635 and the packet does not include a valid MPLS header, the flow sampled packet gets corrupted or truncated. The actual packet is forwarded.

- On QFX10000 Series standalone switches and the QFX Series Virtual Chassis (with QFX3500 and QFX3600 switches), egress firewall filters are not applied to sFlow sampling packets. On these platforms, the software architecture is different from that on other QFX Series devices, and sFlow packets are sent by the Routing Engine (not the line card on the host) and are not transiting the switch. Egress firewall filters affect data packets that are transiting a switch but do not affect packets sent by the Routing Engine. As a result, sFlow sampling packets are always sent to the sFlow collector.

How Adaptive Sampling Works

Every few seconds, or cycle, the sFlow agent collects the interface statistics. From these aggregated statistics, an average number of samples per second is calculated for the cycle. The cycle length depends on the platform:

- Every 12 seconds for EX Series and QFX5K switches and MX Series and PTX Series routers
- Every 5 seconds for QFX Series switches other than QFX5K

If the combined sample rate of all the interfaces on an line card exceeds the adaptive sample rate, a binary backoff algorithm is initiated, which reduces the sample load on the interfaces. Adaptive sampling doubles the sample rate on the affected interfaces, which reduces the sampling load by half. This process is repeated until the CPU load due to sFlow on a given line card comes down to an acceptable level.

Which interfaces on an line card participate in adaptive sampling depends on the platform:

- For MX Series routers and EX Series switches, the sample rates on all the interfaces on the line card are adapted.
- For PTX Series routers and QFX Series switches, only the five interfaces with the highest sample rates on the line card are adapted.

For all platforms, the increased sampling rates remain in effect until one of the following conditions is achieved:

- The device is rebooted.
- A new sample rate is configured.

If you have enabled the adaptive sampling fallback feature and, because of a traffic spike, the number of samples increases to the configured sample-limit-threshold, then the adaptive sampling rate is reversed.

Adaptive Sampling Fallback

The *adaptive sampling fallback* feature, when configured and after adaptive sampling has taken place, uses a binary backup algorithm to decrease the sampling rate (thus, increasing the sampling load) when

the number of samples generated is less than the configured `sample-limit-threshold` value, without affecting normal traffic.

Starting in Junos OS Release 18.3R1, for EX Series switches, Junos OS supports the adaptive sampling fallback feature. Starting in Junos OS Release 19.1R1, for MX Series, PTX Series, and QFX Series devices, Junos OS supports the adaptive sampling fallback feature.

Adaptive sampling fallback is disabled by default. To enable this feature, include the `fallback` and `adaptive-sample-rate` *`sample-limit-threshold`* options in the `[edit protocols sflow adaptive-sample-rate]` hierarchy level.

After adaptive sampling has taken place and the line card is underperforming—that is, the number of samples generated in a cycle are less than the configured value for the `sample-limit-threshold` statement—for five continuous cycles of adaptive sampling, the adapted rate is reversed. If the reverse adaptation has happened and the number of samples generated in a cycle is less than half of the current adapted rate again (and, therefore, for five continuous cycles), another reverse adaptation can happen.

Reverse adaptation does not occur if the interfaces are already at the configured rate.

Adaptive Sampling Limitations

The following are limitations of the adaptive sample feature:

- On standalone routers or standalone QFX Series switches, if you configure sFlow on multiple interfaces and with a high sampling rate, we recommend that you specify a collector that is on the data network instead of on the management network. Having a high volume of sFlow traffic on the management network might interfere with other management interface traffic.
- On routers, sFlow does not support graceful restart. When a graceful restart occurs, the adaptive sampling rate is set to the user-configured sampling rate.
- On a rate-selectable line card (which supports multiple speeds), interfaces with the highest sample count are selected for adaptive sampling fallback. The backup algorithm selects those interfaces on which the adaptive sampling rate is increased the maximum number of times and then decreases the sampling rate on each of those interfaces every five seconds. However, on a single-rate line card, only one sample rate is supported per line card, and the adaptive sampling fallback mechanism backs up the sampling rate on all the interfaces of the line card.

CHAPTER 11

Packet Flow Accelerator Diagnostics Software

IN THIS CHAPTER

- Packet Flow Accelerator Diagnostics Software and Other Utilities Overview | 862
- Install Ethernet and PTP Scripts | 894
- Install Packet Flow Accelerator Diagnostics Software | 897

Packet Flow Accelerator Diagnostics Software and Other Utilities Overview

IN THIS SECTION

- External and Internal Ports and Network Interface Card Ports | 863
- Packet Flow Accelerator Diagnostics Software Tests and Scripts | 864
- Ikondiag Command | 865
- Basic Functionality Tests | 866
- Ethernet Tests and Scripts | 869
- Stress Tests | 876
- PTP Tests | 876
- QFX-PFA-4Q Module LED Tests | 878
- Packet Flow Accelerator Diagnostics Utilities | 879
- Sample Output for Packet Accelerator Diagnostics Software | 886

You can use Packet Flow Accelerator Diagnostics software to validate the integrity of the QFX-PFA-4Q module and the QFX5100-24Q-AA switch. The Packet Flow Accelerator Diagnostics software contains standard diagnostics, orchestration diagnostics, Precision Time Protocol (PTP) and synchronization diagnostics, and other utilities. The Packet Flow Accelerator Diagnostics software runs in a guest virtual

machine (VM) on the QFX5100-24Q-AA switch and requires that you configure guest VM options in the Junos OS CLI.

The QFX-PFA-4Q module contains four 40-Gigabit Ethernet QSFP+ interfaces, an FPGA module, and timing input and output interfaces to support Precision Time Protocol applications. The FPGA module contains logic that you can customize for processing compute-intensive, latency-sensitive, high-volume transactions.

Before you can run the Packet Flow Accelerator Diagnostics software and utilities, make sure you have performed the following tasks:

- Verify that you have installed the QFX-PFA-4Q module installed on the QFX5100-24Q-AA switch. For more information, see Installing an Expansion Module in a QFX5100 Device
- Make sure you have Junos OS Release 14.1X53-D27 with enhanced automation installed on the QFX5100-24Q-AA switch. For more information, see *Installing Software Packages on QFX Series Devices*.
- Install the Packet Flow Accelerator Diagnostics software. For more information, see No Link Title.

External and Internal Ports and Network Interface Card Ports

Packet Flow Accelerator Diagnostics software and utilities validate the data paths between the external and internal ports on the QFX5100-24Q-AA switch and QFX-PFA-4Q module. Figure 33 on page 863 illustrates the names of the ports on the QFX5100-24Q-AA switch and QFX-PFA-4Q module and how they connect.

Figure 33: Ports on the QFX5100-24Q-AA switch and QFX-PFA-4Q module

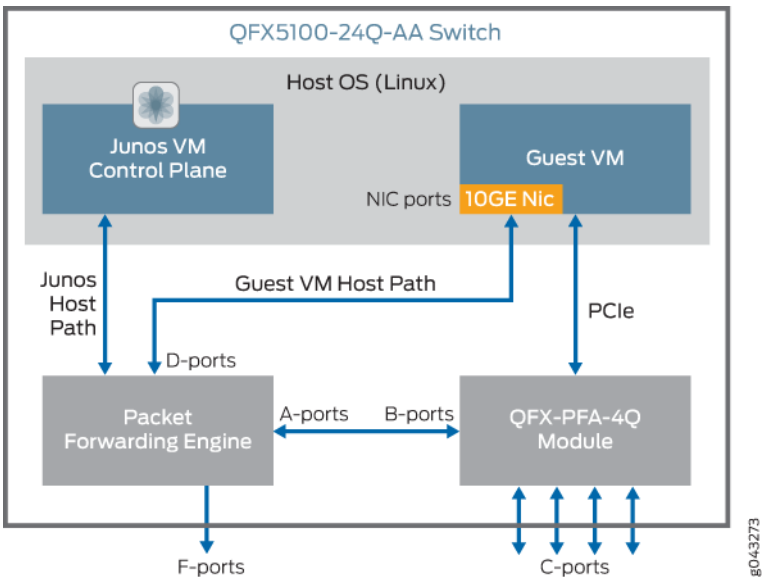


Table 87 on page 864 provides information on the external and internal ports and NIC ports on the QFX5100-24Q-AA switch and QFX-PFA-4Q module.

Table 87: External and Internal Ports on the QFX5100-24Q-AA Switch and the QFX-PFA-4Q Module

A-ports	Interfaces xe-0/0/24 through xe-0/0/39 on the Packet Forwarding Engine (PFE) of the QFX5100-24Q-AA switch connect to the B-ports on the FPGA module on the QFX-PFA-4Q expansion module. A-ports require corresponding B-ports on the FPGA module. You can manage these interfaces through the Junos OS.
B-ports	Internal 10-Gigabit Ethernet ports connect to the FPGA module on the QFX-PFA-4Q module, which then connect to the A-ports on the PFE of the QFX5100-24Q-AA switch. The naming convention for these ports is determined by the guest VM. The guest VM controls the FPGA module.
C-ports	Four, front-facing 40-Gigabit Ethernet ports on the QFX-PFA-4Q module connect to the FPGA module running on the QFX5100-24Q-AA switch and the F-ports on the QFX5100-24Q-AA switch. The guest VM controls the FPGA module.
D-ports	Two 10-Gigabit Ethernet internal ports on the Packet Forwarding Engine of the QFX5100-24Q-AA switch connect to the Ethernet NIC on the QFX5100-24Q-AA switch. The naming convention for these ports is the same one used for the F-ports. You can manage these ports through the Junos OS.
F-ports	Twenty four front-facing 40-Gigabit Ethernet ports on the QFX5100-24Q-AA switch. These ports contain an “et” prefix when in 40-Gigabit Ethernet mode. If you channelize these interfaces, the prefix is “xe.” You can manage these ports through the Junos OS.
NIC ports	Internal interfaces xe-0/0/40 and xe-0/0/41 on the QFX5100-24Q-AA switch connect to the PFE for use on the guest VM. The NIC ports perform the same functions as any other Linux OS NIC port. The NIC ports do not work unless the QFX-PFA-4Q module is installed.

Packet Flow Accelerator Diagnostics Software Tests and Scripts

You can run Packet Flow Accelerator Diagnostics software to test the following subsystems on the QFX-PFA-4Q module:

- FPGA
- QDR SRAM memory
- DRAM memory
- DRAM SPDs

- FPGA-connected PCI Express links
- FPGA-connected Ethernet data (QSFP interfaces)
- QSFP I2C I/O
- PTP I/O

Before you can run any test or script, you need to connect to the console connection of the guest VM. .

The following test sets are available:

- quick-test—Allows you to perform a basic test of all FPGA-attached functionality. These tests take one or two minutes to complete.
- burn-in—Allows you to exercise all FPGA-attached functionality. These tests take several hours to complete.
- individual test mode—Allows you to test a single subsystem with extra configuration options.

Ikondiag Command

To run any of the tests, issue the **ikondiag** command with the following arguments:

> **NOTE**: Before you can run the tests, you need to connect to the console connection of the guest VM.

- -t (quick-test | burn-in | <test name>)

  This argument identifies the test.

- -h

  This argument provides usage details for the test.

- -V

  This argument provides verbose output for the tests.

For example, to run the PTP test, issue the ikondiag -t PTP command at the guest VM prompt:

ikondiag -t PTP

```
[2015-05-07 03:12:20][BEGIN TEST -
PTP]
```

```
************************************************************************

PTP PHY interrupt:
PASS

1G Ethernet PHY packet loopback test:
PASS

PTP clock generation/check:
PASS

UART (ToD) loopback:
PASS

************************************************************************

[2015-05-07 03:13:30][END TEST PTP RESULT
PASS]
```

Basic Functionality Tests

You can test basic functionality on the PCI Express interface and memory components. Table 88 on page 867 lists the names of the tests and their functions.

Table 88: Base Tests

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior
FPGABasic	Tests basic FPGA operation.	Configures the FPGA and reads some simple registers over PCI Express.	None.	quick-test and burn-in	Any failures in this test cause the **ikondiag** command to generate normal test status and error messages, and then to terminate with another error message. You cannot continue testing because all tests rely on the functionality tested by this one.
PCIe	Verifies functionality and stability of bulk transfers of PCIe data.	Repeatedly loops back pseudo random data generated on the CPU to the FPGA and then back to the CPU. Returned data is verified on the CPU.	-i <n> number of repetitions (default = 1 quick-test, 10,000 burn-in) -j <n> size of individual transfer in Mebibytes (default = 100 MiB).	quick-test and burn-in	This test reports erroneous data values and offsets in data transfer. Any failures in this test will cause the ikondiag command to output normal test status and error messages and then terminate with a further error. You cannot continue further testing because all tests rely on functionality tested by this test.

Table 88: Base Tests *(Continued)*

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior
DIMM	Checks SPD query functionality and verifies that correct DIMMs are installed.	Reads data from SPD device on DIMM modules ,reports contents, and checks for erroneous values and verifies: • DIMM part data against expected part data. • SPD temperature is in nominal operating range.	None.	quick-test and burn-in	If any values are unexpected, the test reports erroneous values and provides expected values and ranges.

DRAMMemory	Tests data transfer functionality and stability of FPGA-attached DRAM memory devices.	• Checks that PHYs are initialized correctly.
		• Repeatedly does the following tasks:
		  • Writes to memory from the FPGA
		  • Each pass switches data between: zeros, ones, counter, random, zeros, random, ones, random.
		  • Loops back memory inside the FPGA (simultaneous reads and writes).
		  • Verifies memory from the FPGA
			-i <n> vary number of iterations) default = 1 for quick-test, 500 for burn-in)	quick-test and burn-in	This test reports the number of errors during verification. Number of errors are specified as an accumulated number of errors per byte-lane and DIMM module.

Ethernet Tests and Scripts

The Ethernet tests and scripts test C-ports and traffic between A- and B-ports. The traffic between A- and B-ports is tested by passing the data on the F-ports. For the C-ports, you need to loop back the traffic sent on the C-ports. You can use physical copper loopback cables for this purpose. For the F-ports, , you need to loop back the traffic sent on the F-ports. You can use copper loopback cables for this purpose. Include the F-ports in a VLAN. You can use the python PFAD_exec.py -t 1 script as well as the tests below. The python PFAD_exec.py -t 1 script verifies end-to-end L2 traffic on the external QSFP ports and checks the statistics on the interfaces in Junos OS and the statistics on the interfaces in the Packet Flow Diagnostics software VM. This test will fail if traffic loss is seen on any of the interfaces. There is also a provision to test all the combinations of QSFP ports as well.

Table 89 on page 869 lists the names of the Ethernet tests and their functions. For information on how to install the script, see No Link Title.

Table 89: Ethernet Tests and Scripts

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior
QSFPEthernet	Verifies functionality of Ethernet (QSFP) links.	Generates, receives, and verifies Ethernet frames are at line-rate through the FPGA module. The contents and lengths of packets consist of pseudo-random data. During operation, QSFP connections are all channelized to use 10 Gigabit Ethernet with all 32 Ethernet channels operating in parallel in full-duplex mode.	-i <n> varied number of iterations (default = 1,000 for quick-test, 1e9 for burn-in)	quick-test and burn-in	If the number of packets sent or received correctly are verified as not being equal, this test is considered a failure and the discrepancies between these quantities are reported. This test fails if the external Ethernet connections are not configured for loopback.

Table 89: Ethernet Tests and Scripts *(Continued)*

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior
QSFPI2C	Checks if there is access to the four QSFP modules located on the front of the QFX-PFA-4Q module.	Performs reads of registers in the I2C modules and verifies that the results are as expected. For this test to pass, QSFP media must be inserted into all four ports on the QFX-PFA-4Q module. Any kind of external media can be used (for example, DAC cables, copper loopback, modules, and optical modules).	None.	quick-test and burn-in	This test fails if it cannot detect the presence of a QSFP module or if the values it reads back are unexpected.

Before you can run the Ethernet tests and script successfully, you need to perform the following tasks:

- Externally loop back all of the Ethernet connections (QSFP) on the QFX-PFA-4Q module.

  To loop back the QSFP interfaces on the QFX-PFA-4Q module, attach copper loopback modules on the four QSFP+ interfaces installed on the QFX-PFA-4Q module.

  Attach copper loopback modules on the QSFP+ interfaces (ports 10 through port 13) installed on the QFX5100-24Q-AA switch.

- Channelize ports 10 through 13 on the QFX5100-24Q-AA switch.

- Pair each of the 16 ikonDiag lanes using the equivalent Junos OS interface names with each of the corresponding Junos OS interfaces that were channelized from ports 10 through 13 on the QFX5100-24Q-AA switch.

  NOTE: Each VLAN must be independent contain exactly two associated ports—one 10-Gigabit Ethernet port that is an F-port, and one 10-Gigabit Ethernet port that is an A-port.

  Table 90 on page 871 shows the mappings for the 10-Gigabit Ethernet channels on the QFX-PFA-4Q module F-ports.

Table 90: 10-Gigabit Ethernet Channel Mappings on the QFX-PFA-4Q module F-ports

ikondiag Names	Description
JDFE_XE32_10G	xe-0/0/32
JDFE_XE33_10G	xe-0/0/33
JDFE_XE34_10G	xe-0/0/34
JDFE_XE35_10G	xe-0/0/35
JDFE_XE24_10G	xe-0/0/24
JDFE_XE25_10G	xe-0/0/25
JDFE_XE26_10G	xe-0/0/26
JDFE_XE27_10G	xe-0/0/27
JDFE_XE28_10G	xe-0/0/28
JDFE_XE29_10G	xe-0/0/29
JDFE_XE30_10G	xe-0/0/30
JDFE_XE31_10G	xe-0/0/31
JDFE_XE36_10G	xe-0/0/36
JDFE_XE37_10G	xe-0/0/37
JDFE_XE38_10G	xe-0/0/38

Table 90: 10-Gigabit Ethernet Channel Mappings on the QFX-PFA-4Q module F-ports *(Continued)*

ikondiag Names	Description
JDFE_XE39_10G	xe-0/0/39

Table 91 on page 872 shows the mappings for the 10-Gigabit Ethernet channels on the QFX-PFA-4Q module C-ports.

Table 91: 10-Gigabit Ethernet Channel Mappings on the QFX-PFA-4Q module C-ports

ikondiag Names	Description
JDFE_QSFP0_10G_PORT0	QSFP port #0 10G sub-channel 0
JDFE_QSFP0_10G_PORT1	QSFP port #0 10G sub-channel 1
JDFE_QSFP0_10G_PORT2	QSFP port #0 10G sub-channel 2
JDFE_QSFP0_10G_PORT3	QSFP port #0 10G sub-channel 3
JDFE_QSFP1_10G_PORT0	QSFP port #1 10G sub-channel 0
JDFE_QSFP1_10G_PORT1	QSFP port #1 10G sub-channel 1
JDFE_QSFP1_10G_PORT2	QSFP port #1 10G sub-channel 2
JDFE_QSFP1_10G_PORT3	QSFP port #1 10G sub-channel 3
JDFE_QSFP2_10G_PORT0	QSFP port #2 10G sub-channel 0
JDFE_QSFP2_10G_PORT1	QSFP port #2 10G sub-channel 1
JDFE_QSFP2_10G_PORT2	QSFP port #2 10G sub-channel 2

Table 91: 10-Gigabit Ethernet Channel Mappings on the QFX-PFA-4Q module C-ports *(Continued)*

ikondiag Names	Description
JDFE_QSFP2_10G_PORT3	QSFP port #2 10G sub-channel 3
JDFE_QSFP3_10G_PORT0	QSFP port #3 10G sub channel 0
JDFE_QSFP3_10G_PORT1	QSFP port #3 10G sub-channel 1
JDFE_QSFP3_10G_PORT2	QSFP port #3 10G sub-channel 2
JDFE_QSFP3_10G_PORT3	QSFP port #3 10G sub-channel 3

Table 92 on page 873 provides exact connectivity between the C-ports and A-ports.

Table 92: Exact Connectivity Between C-Ports and A-Ports

QSFP Port Number	Channel Number	Junos OS Interface
QSFP port #0	channel 0	xe-0/0/32
QSFP port #0	channel 1	xe-0/0/33
QSFP port #0	channel 2	xe-0/0/34
QSFP port #0	channel 3	xe-0/0/35
QSFP port #1	channel 0	xe-0/0/24
QSFP port #1	channel 1	xe-0/0/25
QSFP port #1	channel 2	xe-0/0/26
QSFP port #1	channel 3	xe-0/0/27

Table 92: Exact Connectivity Between C-Ports and A-Ports *(Continued)*

QSFP Port Number	Channel Number	Junos OS Interface
QSFP port #2	channel 0	xe-0/0/28
QSFP port #2	channel 1	xe-0/0/29
QSFP port #2	channel 2	xe-0/0/30
QSFP port #2	channel 3	xe-0/0/31
QSFP port #3	channel 0	xe-0/0/36
QSFP port #3	channel 1	xe-0/0/37
QSFP port #3	channel 2	xe-0/0/38
QSFP port #3	channel 3	xe-0/0/39

- Add these interfaces to a VLAN.

Channelize ports 10 through 13 using the Junos CLI.

1. Configure ports 10 through 13 on PIC 1 to operate as 10-Gigabit Ethernet ports.

```
[edit chassis fpc 0 pic 1]
user@switch# set port-range 10 13 channel-speed 10g
```

2. Review your configuration and issue the `commit` command.

```
[edit]
user@switch# commit
commit complete
```

Add the 16 channelized interfaces you just configured to 16 VLANs.

To add the 16 channelized interfaces:

1. Create 16 VLANs.

```
[edit vlans]
user@switch# set v0_0 vlan-id 10
user@switch# set v0_1 vlan-id 11
user@switch# set v0_2 vlan-id 12
user@switch# set v0_3 vlan-id 13
user@switch# set v1_0 vlan-id 14
user@switch# set v1_1 vlan-id 15
user@switch# set v1_2 vlan-id 16
user@switch# set v1_3 vlan-id 17
user@switch# set v2_0 vlan-id 18
user@switch# set v2_1 vlan-id 19
user@switch# set v2_2 vlan-id 20
user@switch# set v2_3 vlan-id 21
user@switch# set v3_0 vlan-id 22
user@switch# set v3_1 vlan-id 23
user@switch# set v3_2 vlan-id 24
user@switch# set v3_3 vlan-id 25
```

2. Add the channelized interfaces to the VLANs.

```
[edit interfaces]
user@switch# set xe-0/0/24 unit 0 family ethernet-switching vlan members v0_0
user@switch# set xe-0/0/25 unit 0 family ethernet-switching vlan members v0_1
user@switch# set xe-0/0/10:0 unit 0 family ethernet-switching vlan members v0_0
user@switch# set xe-0/0/10:1 unit 0 family ethernet-switching vlan members v0_1
user@switch# set xe-0/0/10:2 unit 0 family ethernet-switching vlan members v0_2
user@switch# set xe-0/0/10:3 unit 0 family ethernet-switching vlan members v0_3
user@switch# set xe-0/0/11:0 unit 0 family ethernet-switching vlan members v1_0
user@switch# set xe-0/0/11:1 unit 0 family ethernet-switching vlan members v1_1
user@switch# set xe-0/0/11:2 unit 0 family ethernet-switching vlan members v1_2
user@switch# set xe-0/0/11:3 unit 0 family ethernet-switching vlan members v1_3
user@switch# set xe-0/0/12:0 unit 0 family ethernet-switching vlan members v2_0
user@switch# set xe-0/0/12:1 unit 0 family ethernet-switching vlan members v2_1
user@switch# set xe-0/0/12:2 unit 0 family ethernet-switching vlan members v2_2
user@switch# set xe-0/0/12:3 unit 0 family ethernet-switching vlan members v2_3
user@switch# set xe-0/0/13:0 unit 0 family ethernet-switching vlan members v3_0
user@switch# set xe-0/0/13:1 unit 0 family ethernet-switching vlan members v3_1
```

```
user@switch# set xe-0/0/13:2 unit 0 family ethernet-switching vlan members v3_2
user@switch# set xe-0/0/13:3 unit 0 family ethernet-switching vlan members v3_3
```

3. Review your configuration and issue the `commit` command.

```
[edit]
user@switch# commit
commit complete
```

Stress Tests

The stress tests exercise all high-speed I/Os in parallel. The stress tests require the same external media as you used for the Ethernet tests.Table 93 on page 876 lists the name of the test and its functions.

Table 93: Stress Tests

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior
Stress	Exercises all high-speed I/Os in parallel.	Exercise all of the high-speed I/Os attached to the FPGA in parallel, including: • DRAM • QDR • Ethernet Each test subsystem is exercised in a similar fashion to the individual tests as previously described.	-i <n> varied number of iterations) default = 1 for quick-test, 1,000 for burn-in)	quick-test and burn-in	If any one sub-system fails, the test is stopped. The first sub-system detected to have failed is reported. **NOTE**: If multiple subsystems fail, only the first failed subsystem is reported.

PTP Tests

You can run *PTP* for hardware used with PTP. These tests are helpful if you are creating timing applications. To run the tests, you need to connect SubMiniature version B (SMB) cables, Ethernet loopback cables, and ToD loopback cables for the clocking I/O, ToD serial port, and 1-Gigabit Ethernet connectors. You must connect the SMB, Ethernet, ToD loopback cables between the 10M and PPS output and input connectors. The ToD loopback cable is a standard RJ45 cable with Pin 3 (Tx Data)

connected to Pin 6 (Rx Data). In addition to the PTP tests, you can run scripts included the Packet Flow Accelerator Diagnostics software to test PTP. See Table 95 on page 878 for information on the PTP scripts. The PTP scripts require you to have a Junos OS image with Enhanced Automation installed on the QFX5100-24Q-AA switch. For information on how to install the scripts, see No Link Title.

Table 94 on page 877 lists the names of the PTP tests and their functions:

Table 94: PTP Tests

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior

PTP	Checks functionality of various FPGA-attached time-synchronizing features of the QFX-PFA-4Q module.	Performs various tests on time-synchronizing functionality of the QFX-PFA-4Q module.
		Subtests covered by this test include:
		• Verification of PFE-attached communications.
		• Testing of the PTP PHY
		  • Basic configuration.
		  • FPGA-attached interrupt line.
		  • 1-Gigabit Ethernet loopback (requires external loopback media).
		  • QFX-PFA-4Q module time-syncing related clock generators and feedback routing.
		  • ToD UART port (requires external loop-back media).
			None.	quick-test and burn-in	A failure in any of the subsystems above causes the entire test to fail and generates a report at the end of the test that indicates the pass and fail status of the sub-tests.

Table 95 on page 878 lists the name of the script and its function. This script is not part of the **ikondiag** command. You can run this command Junos OS.

Table 95: PTP Script

Script Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior

./ run_ptp_test	Checks functionality of various FPGA-attached time-syncing features of the QFX-PFA-4Q module.	Performs various tests on time-syncing functionality of the QFX-PFA-4Q module.
		Sub tests covered by this test include:
		• Verification of PFE attached communications.
		• Testing of the PTP PHY
		  • Basic configuration.
		  • FPGA-attached interrupt line.
		  • 1-Gigabith Ethernet loop-back (requires external loop-back media).
		  • QFX-PFA-4Q module time-syncing related clock generators and feedback routing.
		  • ToD UART port (requires external loop-back media).
			None.	None. This test must be run manually.	A failure in any of the subsystems above causes the entire test to fail and generates a report at the end of the test that indicates the pass and fail status of the sub-tests.

QFX-PFA-4Q Module LED Tests

The LED tests examine the LEDs on the QFX-PFA-4Q module.

To run the LED test, issue the **ikon_led_toggle** command. The test might take a few seconds to start because the FPGA is being configured. When you see the message `Toggling LEDs. Send SIGINT (^C) to exit`, the test begins. To terminate the test, type Ctrl-C. Table 96 on page 879 lists the name of the test and its function.

Table 96: QFX-PFA-4Q Module LED Test

Test Name	Description	Details	Optional Arguments	Test Sets	Failure Behavior
ikon_led_toggle	Flashes the LEDs on the QFX-PFA-4Q module for visual inspection.	The following LEDs on the QFX-PFA-4Q module will repeatedly cycle through the following patterns: **NOTE**: The AL and ST LEDs are not included in this test. • The sixteen bicolor LEDs for QSFP status cycle through green, orange, and off. • S0 and S1 LEDs cycle through green and off. • Bottom-left RJ-45 PTP status LED cycles through green, orange, and off. • Bottom-right RJ-45 PTP status LED cycles through green and off rapidly. • Alarm LEDs cycle through orange, red, and off.	None.	None. This test must be run manually.	LEDs might not flash.

Packet Flow Accelerator Diagnostics Utilities

In addition to the Packet Flow Accelerator Diagnostic software tests, there are utilities included in the Packet Flow Accelerator Diagnostics software that you can use to further diagnose issues on the QFX-PFA-4Q module.

NOTE: Before you can run the utilities, you need to connect to the console of the guest VM. For more information on how to access the guest VM, see No Link Title.

Table 97 on page 880 lists the name of the utility and its function.

Table 97: Utilities

Test Name	Descriptions	Details	Expected Output and Behavior
maxtop	Reports FPGA status.	Displays information about the currently configured status of the FPGA module and whether the module is in operation. Checks to make sure very basic driver and FPGA PCI Express link operations are working correctly. If this utility exits with error(s), it is very unlikely that any further FPGA operations will work.	Output should be similar to that shown below. If this output is not shown, there might be a critical failure in the diagnostic software environment, or the PCI Express link to the FPGA is nonfunctional. `MaxTop Tool 2015.1 Found 1 card(s) running MaxelerOS 2015.1 Card 0: QFX-PFA-4Q (P/N: 241124) S/N: 96362301684266423 Mem: 24GB Load average: 0.00, 0.00, 0.00 DFE %BUSY TEMP MAXFILE PID USER TIME COMMAND 0 0.0% - 2fcf249cc7... - - - -`

Table 97: Utilities *(Continued)*

Test Name	Descriptions	Details	Expected Output and Behavior
ikon_snake	Enables snake connectivity between all 10-Gigabit Ethernet channels.	Connects the Rx channel of all 32 x 10-Gigabit Ethernet channels on the FPGA module (QSFP interfaces) to the Tx channel of the respective neighboring connection. This allows all 32 channels to be tested using just a 10-Gigabit Ethernet interface external packet generator, copper loopback modules, and a QSFP <-> 4xSFP breakout cable.	After issuing this test, all Ethernet data will be forwarded after the message `'Snake tool loaded. hit 'enter' to exit.'` is displayed. **NOTE**: During the time before the operating message is printed, the FPGA module might be in the process of being configured, so no data is forwarded. Pressing 'enter' will exit the utility. After the test has finished, packet data continues to be forwarded until another Ethernet utility or test is run.

Table 97: Utilities *(Continued)*

Test Name	Descriptions	Details	Expected Output and Behavior
ikon_eth_util all --digitalloopback	Enables digital-loopback on all 10-Gigabit Ethernet interfaces on the Enables 'snake' connectivity between all QFX-PFA-4Q module 10-Gigabit Ethernet channels.	Connects the Rx side of all 32x 10-Gigabit Ethernet channels on the FPGA module (QSFP) to the Tx side of the same channel.	After issuing this test, all Ethernet data will be forwarded as described after the message `'running press return key to exit'` is displayed. **NOTE**: Before the operating message is displayed, the FPGA module might be in the process of being configured, and no data will be forwarded. Pressing Enter exits the utility. After the test has finished, packet data continues to be forwarded until another Ethernet utility or test is run.

Table 97: Utilities *(Continued)*

Test Name	Descriptions	Details	Expected Output and Behavior
ikon_eth_util	Enables data to pas - through QFX-PFA-4Q module QSFP ports.	Allows data to pass through the QFX-PFA-4Q module QSFP ports on the QFX-PFA-4Q module. **NOTE**: Because all of the QSFP ports are channelized to 10-Gigabit Ethernet, you must use SFP breakout cables when connecting external media.	After issuing this test, all Ethernet data is forwarded as described after the message ‘ running press return key to exit ’ is displayed. **NOTE**: Before the operating message is displayed, the the FPGA module might be in the process of being configured, and no data will be forwarded. Pressing ‘enter’ will exit the utility. After the test has finished, packet data will continue to be forwarded until another Ethernet utility or test is run.

Table 97: Utilities *(Continued)*

Test Name	Descriptions	Details	Expected Output and Behavior

maxnet -v link show	Dumps FPGA packet statistics.	Displays statistics about packets sent and received on all (QSFP) links from the MAC and PHY IP cores in the FPGA. Using the 'v' option provides verbose output.
		
		Here are some important items to note:
		
		• Packet statistics are reset whenever the Altera FPGA is reconfigured; that is, when running different applications that make use of the FPGA.
		
		• The tool only displays data for Ethernet links that are included in the FPGA design. As such, If the FPGA module has not yet been configured, or it is configured with an application that does not use some of the Ethernet links, reduced link details might be displayed.
			Sample output for a single 10-Gigabit Ethernet link is as follows:
			
			`MaxTop Tool 2015.1`
			`Found 1 card(s) running MaxelerOS 2015.1`
			`Card 0: QFX-PFA-4Q (P/N: 241124) S/N: 96362301684266423 Mem: 24GB`
			
			`Load average: 0.00, 0.00, 0.00`
			
			`DFE %BUSY TEMP MAXFILE PID USER TIME COMMAND`
			`0 0.0% - 2fcf249cc7... - - - -`

Table 97: Utilities *(Continued)*

Test Name	Descriptions	Details	Expected Output and Behavior
host2mem I <filename> -o <filename> -t <DDR | QDR0 | QDRPARITY0 | QDR1 | QDRPARITY1>	Writes and then reads arbitrary data from QDR SRAM or DRAM.	Operates by streaming the contents of a binary file to one of the memory resources on the QFX-PFA-4Q module through the FPGA, and then streams the same data back from the memory to another file. **NOTE**: You cannot only read back data from RAM because the contents are not preserved between running multiple tests.	Reports PASSED or FAILED depending on whether the returned data matches the input data.

Table 98 on page 885 lists the command-line arguments for the host2mem utility.

Table 98: Command-Line Arguments

Argument	Description
-- help | -h	Print out usage and exit.
-i <input file>	Input data file.
-o <output file>	Output data file.
-- test | -t <test name>	Test resource. See Table 99 on page 886 for information regarding resources.
-- verbose | -v	Enable verbose mode.

The file format for input and output files is identical. Data is packed consecutively as words based on the width specified in the test mode table below. The size of an input file might be less but must not exceed

the total size of the resource being tested. The size of the output file is the same as the input file and, provided there are no errors, has the same content.

Table 99: File Format Details

Test Mode	Resource	Word Width	Size of Test Data
DDR	DDR SDRAM	192 B	24 GB
QDR0	QDR0 Data	16 B	32 MB
QDRPARITY0	QDR0 Parity bits	2 B	4 MB
QDR1	QDR1 Data	16 B	32 MB
QDRPARITY1	QDR1 Parity bits	2 B	4 MB

The dynamic random-access memory (DRAM) on the QFX-PFA-4Q module contains three dual in-line memory modules (DIMM3, DIMM4, DIMM6), and each data word is split across all three *DIMMs*.Table 100 on page 886 lists the allocation of Bytes to DIMMs.

Table 100: Dual In-Line Memory Modules

0	DIMM3	63	64	DIMM4	127	128	DIMM6	191

Sample Output for Packet Accelerator Diagnostics Software

This section provides some sample output for base tests, Ethernet tests, PTP tests, and utilities.

- ikondiag -t FPGABasic

```
[2015-05-07 03:00:17][BEGIN TEST - FPGABasic]
[2015-05-07 03:00:17][END TEST FPGABasic RESULT PASSED]
```

- ikondiag -t DIMM

```
[2015-05-07 03:01:09][BEGIN TEST - DIMM]
[2015-05-07 03:01:09][END TEST DIMM RESULT PASSED]
```

- ikondiag -t QSFPEthernet

```
[2015-05-07 03:02:33][BEGIN TEST -
QSFPEthernet]

**********************************************

Test
Failed:

  QSFP0_10G_PORT0: FAIL - packets received =
0/1000

  QSFP0_10G_PORT1: FAIL - packets received =
0/1000

  QSFP0_10G_PORT2: FAIL - packets received =
0/1000

  QSFP0_10G_PORT3: FAIL - packets received = 0/1000
  QSFP1_10G_PORT0: FAIL - packets received =
0/1000

  QSFP1_10G_PORT1: FAIL - packets received =
0/1000

  QSFP1_10G_PORT2: FAIL - packets received =
0/1000

  QSFP1_10G_PORT3: FAIL - packets received =
0/1000

  QSFP2_10G_PORT0: FAIL - packets received =
0/1000

  QSFP2_10G_PORT1: FAIL - packets received =
```

```
0/1000

  QSFP2_10G_PORT2: FAIL - packets received =
0/1000

  QSFP2_10G_PORT3: FAIL - packets received =
0/1000

  QSFP3_10G_PORT0: FAIL - packets received = 0/1000
  QSFP3_10G_PORT1: FAIL - packets received =
0/1000

  QSFP3_10G_PORT2: FAIL - packets received =
0/1000

  QSFP3_10G_PORT3: FAIL - packets received =
0/1000

  QSFP4_10G_PORT0: PASS - packets received =
1000/1000

  QSFP4_10G_PORT1: PASS - packets received =
1000/1000

  QSFP4_10G_PORT2: PASS - packets received =
1000/1000

  QSFP4_10G_PORT3: PASS - packets received =
1000/1000

  QSFP5_10G_PORT0: PASS - packets received =
1000/1000

  QSFP5_10G_PORT1: PASS - packets received =
1000/1000

  QSFP5_10G_PORT2: PASS - packets received =
1000/1000

  QSFP5_10G_PORT3: PASS - packets received =
1000/1000

  QSFP6_10G_PORT0: PASS - packets received =
```

```
1000/1000

  QSFP6_10G_PORT1: PASS - packets received =
1000/1000

  QSFP6_10G_PORT2: PASS - packets received =
1000/1000

  QSFP6_10G_PORT3: PASS - packets received =
1000/1000

  QSFP7_10G_PORT0: PASS - packets received =
1000/1000

  QSFP7_10G_PORT1: PASS - packets received =
1000/1000

  QSFP7_10G_PORT2: PASS - packets received =
1000/1000

  QSFP7_10G_PORT3: PASS - packets received =
1000/1000

**********************************************

[2015-05-07 03:02:41][END TEST QSFPEthernet RESULT
PASSED]
```

- ikondiag -t DRAMMemory -i 3

```
[2015-05-07 03:03:37][BEGIN TEST -
DRAMMemory]

[2015-05-07 03:04:21][END TEST DRAMMemory RESULT
PASSED]
```

- ikondiag -t QDRMemory -p -i 3

```
[2015-05-07 03:10:38][BEGIN TEST -
QDRMemory]

[2015-05-07 03:10:45][END TEST QDRMemory RESULT
PASSED]
```

- ikondiag -t Stress -p -i 10

```
[2015-05-07 03:11:24][BEGIN TEST -
Stress]

**********************************************

Test
Failed:

  QSFP0_10G_PORT0: PASS - packets received =
650000/650000

  QSFP0_10G_PORT1: PASS - packets received =
650000/650000

  QSFP0_10G_PORT2: PASS - packets received =
650000/650000

  QSFP0_10G_PORT3: PASS - packets received =
650000/650000

  QSFP1_10G_PORT0: PASS - packets received =
650000/650000

  QSFP1_10G_PORT1: PASS - packets received =
650000/650000

  QSFP1_10G_PORT2: PASS - packets received =
650000/650000

  QSFP1_10G_PORT3: PASS - packets received =
```

```
650000/650000

  QSFP2_10G_PORT0: PASS - packets received = 650000/650000
  QSFP2_10G_PORT1: PASS - packets received = 650000/650000
  QSFP2_10G_PORT2: PASS - packets received =
650000/650000

  QSFP2_10G_PORT3: PASS - packets received =
650000/650000

  QSFP3_10G_PORT0: PASS - packets received =
650000/650000

  QSFP3_10G_PORT1: PASS - packets received =
650000/650000

  QSFP3_10G_PORT2: PASS - packets received =
650000/650000

  QSFP3_10G_PORT3: PASS - packets received =
650000/650000

  QSFP4_10G_PORT0: PASS - packets received =
650000/650000

  QSFP4_10G_PORT1: PASS - packets received =
650000/650000

  QSFP4_10G_PORT2: PASS - packets received = 650000/650000
  QSFP4_10G_PORT3: PASS - packets received =
650000/650000

  QSFP5_10G_PORT0: PASS - packets received =
650000/650000

  QSFP5_10G_PORT1: PASS - packets received =
650000/650000

  QSFP5_10G_PORT2: PASS - packets received =
650000/650000

  QSFP5_10G_PORT3: PASS - packets received =
650000/650000
```

```
  QSFP6_10G_PORT0: PASS - packets received =
650000/650000

  QSFP6_10G_PORT1: PASS - packets received = 650000/650000
  QSFP6_10G_PORT2: PASS - packets received =
650000/650000

  QSFP6_10G_PORT3: PASS - packets received =
650000/650000

  QSFP7_10G_PORT0: PASS - packets received =
650000/650000

  QSFP7_10G_PORT1: PASS - packets received =
650000/650000

  QSFP7_10G_PORT2: PASS - packets received =
650000/650000

  QSFP7_10G_PORT3: PASS - packets received =
650000/650000

**********************************************
```

- ikondiag -t PTP

```
[2015-05-07 03:12:20][BEGIN TEST -
PTP]



**************************************************************************

PTP PHY interrupt:
PASS

1G Ethernet PHY packet loopback test:
PASS

PTP clock generation/check:
PASS
```

```
UART (ToD) loopback:
PASS

**************************************************************************

[2015-05-07 03:13:30][END TEST PTP RESULT
PASS]
```

- ikondiag -t Application -i 2

```
iterations =
2

[2015-05-07 03:14:11][BEGIN TEST - Application
Test]

[2015-05-07 03:17:33][END TEST Application Test RESULT PASSED]
```

- maxtop

```
MaxTop Tool 2015.1
Found 1 card(s) running MaxelerOS 2015.1
Card 0: (P/N: 241124) S/N: 96362301684266423 Mem: 24GB

Load average: 0.00, 0.00, 0.00

DFE  %BUSY  TEMP   MAXFILE        PID    USER       TIME      COMMAND
 0   0.0%   -      7e2198e5c0...  -      -          -         -
```

- ikon_eth_util --all-pass-through

```
Ikon Ethernet Pass Through Utility
setting portConnect_QSFP4_10G_PORT0_QSFP0_10G_PORT0 to 1
setting portConnect_QSFP4_10G_PORT1_QSFP0_10G_PORT1 to 1
setting portConnect_QSFP4_10G_PORT2_QSFP0_10G_PORT2 to 1
setting portConnect_QSFP4_10G_PORT3_QSFP0_10G_PORT3 to 1
setting portConnect_QSFP1_10G_PORT0_QSFP5_10G_PORT0 to 1
```

```
setting portConnect_QSFP1_10G_PORT1_QSFP5_10G_PORT1 to 1
setting portConnect_QSFP1_10G_PORT2_QSFP5_10G_PORT2 to 1
setting portConnect_QSFP1_10G_PORT3_QSFP5_10G_PORT3 to 1
setting portConnect_QSFP2_10G_PORT0_QSFP6_10G_PORT0 to 1
setting portConnect_QSFP2_10G_PORT1_QSFP6_10G_PORT1 to 1
setting portConnect_QSFP2_10G_PORT2_QSFP6_10G_PORT2 to 1
setting portConnect_QSFP2_10G_PORT3_QSFP6_10G_PORT3 to 1
setting portConnect_QSFP3_10G_PORT0_QSFP7_10G_PORT0 to 1
setting portConnect_QSFP3_10G_PORT1_QSFP7_10G_PORT1 to 1
setting portConnect_QSFP3_10G_PORT2_QSFP7_10G_PORT2 to 1
setting portConnect_QSFP3_10G_PORT3_QSFP7_10G_PORT3 to 1
running press return key to exit
```

Install Ethernet and PTP Scripts

IN THIS SECTION

- Install Ethernet and PTP Scripts | **894**

Install Ethernet and PTP Scripts

You can use Ethernet and PTP scripts that are included in the Packet Flow Accelerator Diagnostics software to test Ethernet and PTP functionality. Before you can install the scripts, you need to perform the following tasks:

- Make sure the QFX-PFA-4Q module is installed in the QFX5100 switch. See Installing an Expansion Module in a QFX5100 Device .
- Install Junos OS Release 14.1X53-D27 software or later with enhanced automation for the QFX5100 switch. See *Installing Software Packages on QFX Series Devices* .
- Enable SSH and Telnet services on the switch. See Configuring SSH Service for Remote Access to the Router or Switch and Configuring Telnet Service for Remote Access to a Switch .
- Install the Packet Flow Accelerator Diagnostics Software. See No Link Title .

To install the scripts:

1. Log into the guest VM using the `request app-engine virtual-machine-shell` *guest-VM-name*. The maximum length for the guest VM name is 255 characters. Make sure you are logged in as root when you enter this command.

```
root>  request app-engine virtual-machine-shell diagnostics
```

2. Enter a valid username and password combination for the guest VM.
3. Enter the **guest-util diag-install** ***guest VM IP address*** command at the shell prompt.

   Use the same IP address you used for configuring the local management address for the guest VM.

```
[root@localhost ~] guest-util diag-install 192.168.1.10
```

4. Change directories to /var/tmp to edit the PFAD_params.cfg file.

```
[root@localhost ~] cd /var/tmp
```

5. Open the PFAD_params.cfg file using an editor of your choice.

   Here is an example of what is contained in the file:

```
[params]


# log level
LOGLEVEL = 'TRACE'


# my variables
VLAN1_NAME      = 'VLAN100'
VLAN1_ID        = '100'
JUNOS_USERNAME  = 'test'
ROOT_USERNAME   = 'root'
JUNOS_PSWD      = 'juniper123'
GUEST_PSWD      = 'diag'
ROOT_PSWD       = 'root123'



# my duts
DUTS = {
    'R0': "10.204.43.170",
```

```
    }



TOPOLOGY = 'IF1 = 'et-0/0/2’
                                IF2 = ‘et-0/0/3’

PFAD_params.cfg: unmodified: line 1
```

6. Configure the management IP address.

```
                                DUTS = {
                                                ‘R0':
"10.204.43.170”,                                                                }
```

7. Configure the PTP interfaces.

   IF1 is the primary source, and IF2 is the secondary source.

   Configure IF1 as et-0/0/2, and IF2 as et-0/0/3.

```
                                IF1 = '2’  <<<<< Change it
                                IF2 = ‘3’  <<<<< Change it
```

8. Save the changes you made to the PFAD_params.cfg file.
9. Run the scripts by issuing one of the following commands at the guest VM prompt.
   - To test traffic orchestration:

     python PFAD_exec.py -t 1

   - To test PTP:

     ./run_ptp_test

   - To test Broadsync:

     ./run_broadsync_test

Install Packet Flow Accelerator Diagnostics Software

IN THIS SECTION

- Packet Flow Accelerator Diagnostics Software Overview | **897**
- Verify That the QFX-PFA-4Q Expansion Module Is Installed | **898**
- Download the Packet Flow Diagnostics Software | **898**
- Copy the Packet Flow Diagnostics Software Package to the Switch | **899**
- Install the Packet Flow Diagnostics Software on the Switch | **899**
- Configure the Guest VM Options to Launch the Guest VM on the Host | **901**
- Verify That the Guest VM is Working | **904**
- Access the Guest VM | **905**
- Verify That the FPGA Module Is Working | **907**
- Validate the Connections Between QFX5100-24Q-AA Switch Network Ports and QFX-PFA-4Q Module Ports | **908**
- Uninstall the Guest VM | **911**

Packet Flow Accelerator Diagnostics Software Overview

You can use Packet Flow Accelerator Diagnostics software to test the FPGA module in the QFX-PFA-4Q module installed on the QFX5100-24Q-AA switch as well as the data paths between the FPGA module and the QFX5100-24Q-AA switch. The Packet Flow Accelerator Diagnostics software contains standard diagnostics, orchestration diagnostics, and Precision Time Protocol (PTP) and synchronization diagnostics. In addition to the Packet Flow Accelerator Diagnostics software tests, there are utilities included in the Packet Flow Accelerator Diagnostics software that you can use to further diagnose issues on the QFX-PFA-4Q module. For information on how to install the QFX-PFA-4Q module, see Installing an Expansion Module in a QFX5100 Device.

To run the orchestration diagnostics, PTP and synchronization diagnostics, and utilities contained in the Packet Flow Accelerator Diagnostics software, you need to have a Junos OS Release 14.1X53-D27 software or later with enhanced automation installed on your QFX5100 switch. For information on how to download and install Junos OS software, see *Installing Software Packages on QFX Series Devices*.

The Packet Flow Accelerator Diagnostics software runs in a guest VM on the switch and requires that you configure guest VM options in the Junos OS CLI.

Verify That the QFX-PFA-4Q Expansion Module Is Installed

Before you install the Packet Flow Accelerator Diagnostics software, verify that the QFX-PFA-4Q module is installed.

From the CLI prompt, issue the `show chassis hardware` command.

```
{master:0}
root> show chassis hardware
Hardware inventory:
Item             Version  Part number  Serial number     Description
Chassis                                VX3715020024      QFX5100-24Q-AA
Pseudo CB 0
Routing Engine 0          BUILTIN      BUILTIN           QFX Routing Engine
FPC 0            REV 02   650-057155   VX3715020024      QFX5100-24Q-AA
  CPU                     BUILTIN      BUILTIN           FPC CPU
  PIC 0                   BUILTIN      BUILTIN           24x 40G-QSFP-AA
    Xcvr 6       REV 01   740-032986   QD334902          QSFP+-40G-SR4
  PIC 1          REV 01   711-060247   VY3115060052      QFX-PFA-4Q
Power Supply 0   REV 03   740-041741   1GA24082731       JPSU-650W-AC-AFO
Power Supply 1   REV 03   740-041741   1GA24082726       JPSU-650W-AC-AFO
Fan Tray 0                                               QFX5100 Fan Tray 0, Front to Back
Airflow - AFO
Fan Tray 1                                               QFX5100 Fan Tray 1, Front to Back
Airflow - AFO
Fan Tray 2                                               QFX5100 Fan Tray 2, Front to Back
Airflow - AFO
Fan Tray 3                                               QFX5100 Fan Tray 3, Front to Back
Airflow - AFO
Fan Tray 4                                               QFX5100 Fan Tray 4, Front to Back
Airflow - AFO
```

From the CLI output, you can see that the four QSFP+ interfaces (4x40G QSFP+) contained in the QFX-PFA-4Q module. are installed.

Download the Packet Flow Diagnostics Software

NOTE: To access the download site, you must have a service contract with Juniper Networks and an access account. If you need help obtaining an account, complete the registration form at the Juniper Networks website

https://www.juniper.net/registration/Register.jsp .

To download the Packet Flow Diagnostics software package from the Juniper Networks Support website, go to https://www.juniper.net/support/ :

1. Using a Web browser, navigate to https://www.juniper.net/support .
2. Click **Download Software**.
3. In the Switching box, click **Junos OS Platforms**.
4. In the QFX Series section, click the name of the platform for which you want to download software.
5. Click the Software tab and select the release number from the Release drop-down list.
6. In the Install Package section on the Software tab, select the Install Package for the release.

   A login screen appears.
7. Enter your name and password and press Enter.
8. Read the End User License Agreement, click the `I agree` radio button, and then click `Proceed`.
9. Save the `pfadiag_vm-rXXXXX.img.gz` file on your computer.
10. Open or save the Packet Flow Diagnostics software package either to the local system in the **var/tmp** directory or to a remote location. If you are saving the installation package to a remote system, make sure that you can access it using HTTP, TFTP, FTP, or scp.

Copy the Packet Flow Diagnostics Software Package to the Switch

To copy the packet flow diagnostics software package to the switch:

Copy the packet flow diagnostics package to the switch using any file transfer protocol:

For example:

```
root% scp //hostname/pathname/pfadiag_vm-rXXXXX.img.gz /var/tmp
```

Install the Packet Flow Diagnostics Software on the Switch

To install the packet flow diagnostics software package on the switch:

1. Install the Packet Flow Diagnostics software on the switch.

   This might take a few minutes.

   If the Packet Flow Diagnostics software resides locally on the switch, issue the following command:

   ```
   {master:0}
   root> request system software add virtual-machine-package /var/tmp/pfadiag_vm-rXXXXX.img.gz
   ```

```
Installing virtual-machine package..
Copying virtual-machine package..
Uncompressing virtual-machine package..
Finished virtual-machine package installation.
```

2. Issue the `show version` command to verify that the installation was successful.

```
{master:0}
root> show version
fpc0:

--------------------------------------------------------------------------

Hostname:
switch

Model: qfx5100-24q-
aa

Junos: 14.1X53-
D27_vjunos.62

JUNOS Base OS Software Suite [14.1X53-
D26_vjunos.62]

JUNOS Base OS boot [14.1X53-
D27_vjunos.62]

JUNOS Crypto Software Suite [14.1X53-
D27_vjunos.62]

JUNOS Online Documentation [14.1X53-
D27_vjunos.62]

JUNOS Kernel Software Suite [14.1X53-
D27_vjunos.62]

JUNOS Packet Forwarding Engine Support (qfx-ex-x86-32) [14.1X53-
D27_vjunos.62]

JUNOS Routing Software Suite [14.1X53-
```

```
D27_vjunos.62]

JUNOS Enterprise Software Suite [14.1X53-
D27_vjunos.62]

JUNOS py-base-i386 [14.1X53-
D27_vjunos.62]

JUNOS py-extensions-i386 [14.1X53-
D27_vjunos.62]

JUNOS Host Software [14.1X53-
D27_vjunos.62]

Junos for Automation
Enhancement

JUNOS GUEST-VM Software [pfadiag_vm-rXXXXX-
ve]



{master:0}
```

The CLI output shows that the Packet Flow Accelerator Diagnostics software was installed.

Configure the Guest VM Options to Launch the Guest VM on the Host

To configure the guest VM options:

1. Configure the following options for guest *VM* support in the Junos OS CLI at the [edit] hierarchy.
   - Compute cluster name
   - Compute node name
   - VM instance name
   - Dedicated management interface for guest VM
   - Third-party package name
   - Internal IP address of the guest VM
2. Configure the name of the compute cluster and compute node.

The name of the compute cluster must be default-cluster, and the name of the name of the compute node must be default-node; otherwise, launching the guest VM fails.

```
{master:0}
root# set services app-engine compute-cluster default-cluster compute-node default-node
hypervisor
```

3. Configure the name of the VM instance and the name of the third party application.

```
{master:0}
root# set services app-engine virtual-machines instance instance-name package package-name
```

> **NOTE**: The package names in the `show app-engine virtual-machine-package` command and the `show version` command should match.

```
{master:0}
root# set services app-engine virtual-machines instance diagnostics package pfadiag_vm-rXXXXX-
ve
```

4. Associate the VM instance with the configured compute cluster and compute node.

```
{master:0}
root# set services app-engine virtual-machines instance instance-name compute-cluster name
compute-node name
```

```
{master:0}
root# set services app-engine virtual-machines instance diagnostics compute-cluster default-
cluster compute-node default-node
```

> **NOTE**: The name of the compute cluster must be default-cluster, and the name of the compute node must be default-node; otherwise, launching the guest VM fails.

5. Configure the local management IP address.

   This IP address is used for the internal bridging interface. The host uses this IP address to check the availability of the guest VM.

NOTE: Do not use 192.168.1.1 and 192.168.1.2 as IP addresses because they are used by the Host-OS and Junos OS respectively.

```
{master:0}
root# set services app-engine virtual-machines instance instance-name local-management family
inet address 192.168.1.X
```

```
{master:0}
root# set services app-engine virtual-machines instance diagnostics local-management family
inet address 192.168.1.10
```

6. Configure the management interface for the guest VM.

   This management interface is separate from the one used for Junos OS.

```
{master:0}
root # set services app-engine virtual-machines instance diagnostics management-interface em1
```

NOTE: The management interface name must be either em0 or em1. The configuration will fail if you do not configure a management interface and then commit the configuration.

   The new management interface is provisioned for the guest VM.

7. Commit the configuration.

```
{master:0}
root# commit
```

   Here are the results of the configuration:

```
services {
    app-engine {
        compute-cluster default-cluster {
            compute-node default-node {
                hypervisor;
            }
```

```
        }
        virtual-machines {
            instance diagnostics {
                package pfadiag_vm-rXXXXX-ve;
                local-management {
                    family inet {
                        address 192.168.1.10;
                    }
                }
                compute-cluster default-cluster {
                    compute-node default-node;
                }
                management-interface em1;
            }
        }
    }
}
```

Verify That the Guest VM is Working

To verify that the guest VM is working:

Issue the following `show` commands to verify that everything is working correctly:

- root> **show app-engine status**

```
Compute cluster: default-cluster
  Compute Node: default-node, Online
```

  The status should be Online.

- root> **show app-engine virtual-machine instance**

```
VM name                 Compute cluster          VM status
diagnostics              default-cluster          ACTIVE
```

  The VM status should be active.

- root> **show app-engine virtual-machine package**

```
VM package: pfadiag_vm-rXXXXX-ve
Compute cluster                  Package download status
default-cluster                  DOWNLOADED
```

Access the Guest VM

To access the guest VM:

1. Log into the guest VM.
   - Specify the guest VM name using the `request app-engine virtual-machine-shell` *`guest-VM-name`* command. The maximum length for the guest VM name is 255 characters. Make sure you are logged in as root when you enter this command.

     ```
     root> request app-engine virtual-machine-shell diagnostics
     ```

   - Enter a valid username and password combination for the guest VM.

     NOTE: The first time you log in, the username is root. There is no password. After you log in, you will be prompted to create a password.

     For example:

     ```
     Maxeler Ikon Diagnostics VM r44702

     diagnostics login: root
     You are required to change your password immediately (root enforced)
     New password:
     Retype new password:
     ```

2. Issue the `ifconfig -a` command to see the names of the management interface that is used to access the guest VM from outside of the network, name of the management interface that is used for internal use, and the NIC ports used in the diagnostics VM.

   In this example, the `heartbeat` address is the IP address that is used for internal use , the `management` interface is used for external communications, and the xe-0/0/40 and xe-0/0/41 interfaces are the NIC ports used in the diagnostics VM. The `heartbeat` is configured by default. The IP address of the `heartbeat` is the same as the IP address you configured for Junos OS.

You can associate one of the interfaces to the guest VM by issuing the `set services app-engine virtual-machines instance` *`name`* `management-interface` *`interface-name`*. command. Use the same IP address as the one you configured using the `set services app-engine virtual-machines instance test local-management family inet address 192.168.1.10`. The MAC addresses associated with these interfaces are used for internal bridging.

```
[root@ikondiag ~]# ifconfig -a
heartbeat Link encap:Ethernet  HWaddr 52:54:00:5D:DB:01
          inet addr:192.168.1.10  Bcast:0.0.0.0  Mask:255.255.255.0
          inet6 addr: fe80::5054:ff:fe5d:db01/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:282 errors:0 dropped:0 overruns:0 frame:0
          TX packets:266 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:24955 (24.3 KiB)  TX bytes:24232 (23.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

management Link encap:Ethernet  HWaddr 52:54:00:76:B3:C4
          inet6 addr: fe80::5054:ff:fe76:b3c4/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:6 errors:0 dropped:0 overruns:0 frame:0
          TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:438 (438.0 b)  TX bytes:1836 (1.7 KiB)

xe-0-0-40 Link encap:Ethernet  HWaddr EA:8B:BB:75:56:FE
          inet6 addr: fe80::e88b:bbff:fe75:56fe/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:140 (140.0 b)

xe-0-0-41 Link encap:Ethernet  HWaddr 3E:1A:00:94:ED:5B
```

```
          inet6 addr: fe80::3c1a:ff:fe94:ed5b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 b)  TX bytes:230 (230.0 b)
```

Verify That the FPGA Module Is Working

You can use the following utilities to verify that the FPGA module on the QFX-PFA-4Q module is working.

To verify that the FPGA module is working:

1. Issue the **lspci |grep "RAM memory"** command at the guest VM login prompt.

```
[root@ikondiag ~]# lspci |grep "RAM memory"
00:09.0 RAM memory: Juniper Networks Device 0078
```

   The output shows that Juniper Networks Device 0078 is working.

2. Issue the **lspci |grep Co-processor** command at the guest VM login prompt:

```
[root@ikondiag ~]# lspci |grep Co-processor
:0a.0 Co-processor: Maxeler Technologies Ltd. Device 0006
```

   The output shows that Maxeler Technologies Ltd. Device 0006 is working.

3. Issue the **maxtop** command at the guest VM login prompt:

   NOTE: If there are errors in the command output, relaunch the guest VM.

```
[root@ikondiag ~]# maxtop
```

```
MaxTop Tool 2015.1
Found 1 card(s) running MaxelerOS 2015.1
Card 0: QFX-PFA-4Q (P/N: 241124) S/N: 96362301684266423 Mem: 24GB
```

```
Load average: 0.00, 0.00, 0.00

DFE  %BUSY  TEMP   MAXFILE        PID    USER       TIME      COMMAND
 0   0.0%   -      2fcf249cc7...  -      -          -         -
```

Validate the Connections Between QFX5100-24Q-AA Switch Network Ports and QFX-PFA-4Q Module Ports

You can use the **ikon_eth_util –all-pass-through** utility to validate the connections between the QFX5100-24Q-AA switch network ports and the QFX-PFA-4Q module ports.

In this example, the **ikon_eth_util --all-pass-through** utility will validate the following connections between the F-ports, A-ports, B-ports, and C-ports. Table 101 on page 908 provides the ports that are validated in this example.

Table 101: Validating Ports

F-Ports	A-Ports	B-Ports	C-Ports
xe-0/0/10:2 This interface is one of the 10-Gigabit Ethernet ports on the QFX5100-24Q-AA switch. You can manage these ports through the Junos OS.	xe-0/0/32 This interface connects the PFE of the QFX5100-24Q-AA switch to the B-ports on the FPGA module on the QFX-PFA-4Q module.	JDFE_XE32_10G This interface is an Internal 10-Gigabit Ethernet port on the FPGA module on the QFX-PFA-4Q module and connects to the A-ports on the PFE of the QFX5100-24Q-AA switch.	JDFE_QSFP0_10G_PORT0 [External Port 0-0] This interface is one of the front-facing 40-Gigabit Ethernet ports on the QFX-PFA-4Q module and connects to the guest VM running on the QFX5100-24Q-AA switch and the F-ports on the QFX5100-24Q-AA switch.

To validate the connections between the QFX5100-24Q-AA switch network ports and the QFX-PFA-4Q module ports:

1. Configure a VLAN and VLAN ID:

```
[edit vlans]
user@switch # set VLAN_TEST vlan-id 100
```

2. Associate the F-port and A-port in this VLAN so that the FPGA and PFE can communicate:

```
[edit interfaces]
user@switch # set xe-0/0/10:2 unit 0 family ethernet-switching vlan members VLAN_TEST
user@switch # set xe-0/0/32 unit 0 family ethernet-switching vlan members VLAN_TEST
```

3. Commit the configuration:

```
[edit]
user@switch # commit synchronize
```

4. Verify that the VLAN has been created.

```
[edit]
user@switch # run show vlans
    Routing instance        VLAN name             Tag          Interfaces
    default-switch          VLAN_TEST              100
                                                               xe-0/0/10:2.0*
                                                               xe-0/0/32.0*
    default-switch          default               1
```

5. Issue the **ikon_eth_util --all-pass-through** command at the guest VM login prompt:

```
[root@ikondiag ~]# ikon_eth_util --all-pass-through
Ikon Ethernet Pass Through Utility
setting portConnect_JDFE_QSFP0_10G_PORT0_JDFE_XE32_10G to 1
setting portConnect_JDFE_QSFP0_10G_PORT1_JDFE_XE33_10G to 1
setting portConnect_JDFE_QSFP0_10G_PORT2_JDFE_XE34_10G to 1
setting portConnect_JDFE_QSFP0_10G_PORT3_JDFE_XE35_10G to 1
setting portConnect_JDFE_XE24_10G_JDFE_QSFP1_10G_PORT0 to 1
setting portConnect_JDFE_XE25_10G_JDFE_QSFP1_10G_PORT1 to 1
setting portConnect_JDFE_XE26_10G_JDFE_QSFP1_10G_PORT2 to 1
setting portConnect_JDFE_XE27_10G_JDFE_QSFP1_10G_PORT3 to 1
setting portConnect_JDFE_XE28_10G_JDFE_QSFP2_10G_PORT0 to 1
setting portConnect_JDFE_XE29_10G_JDFE_QSFP2_10G_PORT1 to 1
setting portConnect_JDFE_XE30_10G_JDFE_QSFP2_10G_PORT2 to 1
setting portConnect_JDFE_XE31_10G_JDFE_QSFP2_10G_PORT3 to 1
setting portConnect_JDFE_XE36_10G_JDFE_QSFP3_10G_PORT0 to 1
setting portConnect_JDFE_XE37_10G_JDFE_QSFP3_10G_PORT1 to 1
```

```
setting portConnect_JDFE_XE38_10G_JDFE_QSFP3_10G_PORT2 to 1
setting portConnect_JDFE_XE39_10G_JDFE_QSFP3_10G_PORT3 to 1
running press return key to exit
```

6. Send traffic to xe-0/0/10:2 on the QFX5100-24Q-AA switch and receive traffic on the front panel port 0-0 on the QFX-PFA-4Q module.
7. Send traffic to the front panel port 0-0 on the QFX-PFA-4Q module and receive traffic on xe-0/0/10:2 on the QFX5100-24Q-AA switch.
8. Verify the statistics for the xe-0/0/10:2 and xe-0/0/32 interfaces by issuing the `show interfaces xe-0/0/10:2 extensive` and `show interfaces xe-0/0/32 extensive` commands.
9. Verify the statistics for the JDFE_XE32_10G and JDFE_QSFP0_10G_PORT0 interfaces by issuing the `maxnet link` commands at the guest VM prompt for the Packet Flow Accelerator Diagnostics software.

   [root@ikondiag ~]# **maxnet link show JDFE_XE32_10G**

```
JDFE_XE32_10G:
         Link Up: true
     MAC address: 00:11:22:33:44:55
      RX Enabled: true
       RX Frames: 1 ok
                  0 error
                  0 CRC error
                  0 invalid/errored
                  1 total
      TX Enabled: true
       TX Frames: 0 ok
                  0 error
                  0 CRC error
                  0 invalid/errored
                  0 total
```

   [root@ikondiag ~]# **maxnet link show JDFE_QSFP0_10G_PORT0**

```
JDFE_QSFP0_10G_PORT0:
         Link Up: true
     MAC address: 00:11:22:33:44:55
      RX Enabled: true
       RX Frames: 0 ok
                  0 error
                  0 CRC error
                  0 invalid/errored
                  0 total
```

```
      TX Enabled: true
       TX Frames: 1 ok
                  0 error
                  0 CRC error
                  0 invalid/errored
                  1 total
```

Uninstall the Guest VM

To remove the guest VM:

1. Delete the configuration statements and uninstall the Packet Flow Accelerator Diagnostics software package.

   For example, to remove the `app-engine` statement:

```
root # delete services app-engine
```

2. Commit the configuration.

```
root# commit
```

3. (Optional) Issue the `show version` command to learn the name of the Packet Flow Accelerator Diagnostics software package.

```
{master:0}
root> show version
fpc0:

--------------------------------------------------------------------------

Hostname:
switch

Model: qfx5100-24q-
aa

Junos: 14.1X53-
D27_vjunos.62

JUNOS Base OS Software Suite [14.1X53-
D27_vjunos.62]
```

```
JUNOS Base OS boot [14.1X53-
D27_vjunos.62]

JUNOS Crypto Software Suite [14.1X53-
D27_vjunos.62]

JUNOS Online Documentation [14.1X53-
D27_vjunos.62]

JUNOS Kernel Software Suite [14.1X53-
D27_vjunos.62]

JUNOS Packet Forwarding Engine Support (qfx-ex-x86-32) [14.1X53-
D26_vjunos.62]

JUNOS Routing Software Suite [14.1X53-
D27_vjunos.62]

JUNOS Enterprise Software Suite [14.1X53-
D27_vjunos.62]

JUNOS py-base-i386 [14.1X53-
D27_vjunos.62]

JUNOS py-extensions-i386 [14.1X53-
D27_vjunos.62]

JUNOS Host Software [14.1X53-
D27_vjunos.62]

Junos for Automation
Enhancement

JUNOS GUEST-VM Software [pfadiag_vm-rXXXXX-
ve]



{master:0}
```

4. Issue the `request system software delete virtual-machine-package <package-name>` command to uninstall the Packet Flow Accelerator Diagnostics software.

```
root> request system software delete virtual-machine-package pfadiag_vm-rXXXXX-ve
fpc0:
--------------------------------------------------------------------------
Deleted virtual-machine package dpfadiag_vm-rXXXXX-ve ...
```

PART 7

Monitoring Common Security Features

Display Real-Time Information from Device to Host | 915

Monitor Security Policies | 922

Monitor Interfaces and Switching Functions | 923

Display Real-Time Information from Device to Host

SUMMARY

This section describes how to display real-time monitoring information about each device between the device and a destination host.

IN THIS SECTION

- Display Real-Time Monitoring Information | 915
- Display Multicast Path Information | 918

Display Real-Time Monitoring Information

To display real-time monitoring information about each device between the device and a specified destination host, enter the `traceroute monitor` command with the following syntax:

```
user@host> traceroute monitor host <count number> <inet | inet6> <interval seconds> <no-resolve>
<size bytes><source source-address> <summary>
```

Table 102 on page 915 describes the `traceroute monitor` command options.

Table 102: CLI traceroute monitor Command Options

Option	Description
host	Sends traceroute packets to the hostname or IP address you specify.
`count` *number*	(Optional) Limits the number of ping requests, in packets, to send in summary mode. If you do not specify a count, ping requests are continuously sent until you press Q.
`inet`	(Optional) Forces the traceroute packets to an IPv4 destination.
`inet6`	(Optional) Forces the traceroute packets to an IPv6 destination.

Table 102: CLI traceroute monitor Command Options *(Continued)*

Option	Description
`interval` *seconds*	(Optional) Sets the interval between ping requests, in seconds. The default value is `1` second.
`no-resolve`	(Optional) Suppresses the display of the hostnames of the hops along the path.
`size` *bytes*	(Optional) Sets the size of the ping request packet. The size can be from `0` through `65,468` bytes. The default packet size is `64` bytes.
`source` *address*	(Optional) Uses the source address that you specify, in the traceroute packet.
`summary`	(Optional) Displays the summary traceroute information.

To quit the `traceroute monitor` command, press Q.

The following is sample output from a `traceroute monitor` command:

```
user@host> traceroute monitor host2
```

```
                                                  My traceroute  [v0.69]
 host (0.0.0.0)(tos=0x0 psize=64
bitpattern=0x00)                                                  Wed Mar 14 23:14:11
2007
 Keys:  Help   Display mode   Restart statistics   Order of fields   quit

Packets               Pings
  Host                                                                        Loss
%   Snt   Last   Avg  Best  Wrst StDev
  1. 173.24.232.66
0.0%     5    9.4   8.6   4.8   9.9   2.1
  2. 173.24.232.66
0.0%     5    7.9  17.2   7.9  29.4  11.0
  3. 173.24.232.66
0.0%     5    9.9   9.3   8.7   9.9   0.5
```

```
   4. 173.24.232.66
 0.0%     5    9.9   9.8   9.5  10.0   0.2
```

Table 103 on page 917 summarizes the output fields of the display.

Table 103: CLI traceroute monitor Command Output Summary

Field	Description
`host`	Hostname or IP address of the device issuing the `traceroute monitor` command.
`psize`*size*	Size of ping request packet, in bytes.
Keys	
`Help`	Displays the Help for the CLI commands. Press H to display the Help.
`Display mode`	Toggles the display mode. Press D to toggle the display mode
`Restart statistics`	Restarts the `traceroute monitor` command. Press R to restart the `traceroute monitor` command.
`Order of fields`	Sets the order of the displayed fields. Press O to set the order of the displayed fields.
`quit`	Quits the `traceroute monitor` command. Press Q to quit the `traceroute monitor` command.
Packets	
number	Number of the hop (device) along the route to the final destination host.
`Host`	Hostname or IP address of the device at each hop.

Table 103: CLI traceroute monitor Command Output Summary *(Continued)*

Field	Description
Loss%	Percent of packet loss. The number of ping responses divided by the number of ping requests, specified as a percentage.
Pings	
Snt	Number of ping requests sent to the device at this hop.
Last	Most recent round-trip time, in milliseconds, to the device at this hop.
Avg	Average round-trip time, in milliseconds, to the device at this hop.
Best	Shortest round-trip time, in milliseconds, to the device at this hop.
Wrst	Longest round-trip time, in milliseconds, to the device at this hop.
StDev	Standard deviation of round-trip times, in milliseconds, to the device at this hop.

Display Multicast Path Information

To display information about a multicast path from a source to the device, enter the `mtrace from-source` command with the following syntax:

```
user@host> mtrace from-source source host <extra-hops number> <group address> <interval seconds>
<max-hops number> <max-queries number> <response host> <routing-instance routing-instance-name>
<ttl number> <wait-time seconds> <loop> <multicast-response | unicast-response> <no-resolve> <no-
router-alert> <brief | detail>
```

Table 104 on page 919 describes the `mtrace from-source` command options.

Table 104: CLI mtrace from-source Command Options

Option	Description
`source` *host*	Traces the path to the specified hostname or IP address.
`extra-hops` *number*	(Optional) Sets the number of extra hops to trace past nonresponsive devices. Specify a value from `0` through `255`.
`group` *address*	(Optional) Traces the path for the specified group address. The default value is `192.0.2.0`.
`interval` *seconds*	(Optional) Sets the interval between statistics gathering. The default value is `10`.
`max-hops` *number*	(Optional) Sets the maximum number of hops to trace toward the source. Specify a value from `0` through `255`. The default value is `32`.
`max-queries` *number*	(Optional) Sets the maximum number of query attempts for any hop. Specify a value from `1` through `32`. The default value is `3`.
`response` *host*	(Optional) Sends the response packets to the specified hostname or IP address. By default, the response packets are sent to the device.
`routing-instance` *routing-instance-name*	(Optional) Traces the routing instance you specify.
`ttl` *number*	(Optional) Sets the time-to-live (TTL) value in the IP header of the query packets. Specify a hop count from `0` through `255`. The default value for local queries to the *all routers* multicast group is `1`. Otherwise, the default value is `127`.
`wait-time` *seconds*	(Optional) Sets the time to wait for a response packet. The default value is `3` seconds.
`loop`	(Optional) Loops indefinitely, displaying rate and loss statistics. To quit the `mtrace` command, press Ctrl-C.

Table 104: CLI mtrace from-source Command Options *(Continued)*

Option	Description
`multicast-response`	(Optional) Forces the responses to use multicast.
`unicast-response`	(Optional) Forces the response packets to use unicast.
`no-resolve`	(Optional) Does not display hostnames.
`no-router-alert`	(Optional) Does not use the device alert IP option in the IP header.
`brief`	(Optional) Does not display packet rates and losses.
`detail`	(Optional) Displays packet rates and losses if a group address is specified.

The following is sample output from the `mtrace from-source` command:

```
user@host> mtrace from-source source 192.1.4.1 group 224.1.1.1
```

```
Mtrace from 192.1.4.1 to 192.1.30.2 via group 224.1.1.1 Querying full reverse path... * *
0  ? (192.1.30.2)  -1  ? (192.1.30.1)  PIM  thresh^ 1    -2  routerC.mycompany.net (192.1.40.2)
PIM  thresh^ 1    -3  hostA.mycompany.net (192.1.4.1) Round trip time 22 ms; total ttl of 2
required.  Waiting to accumulate statistics...Results after 10 seconds:    Source
Response Dest    Overall     Packet Statistics For Traffic From 192.1.4.1  192.1.30.2
Packet      192.1.4.1 To 224.1.1.1      v       __/  rtt   16 ms     Rate       Lost/Sent = Pct
Rate 192.168.195.37  192.1.40.2      routerC.mycompany.net       v     ^      ttl
2                      0/0    = --    0 pps 192.1.40.1      192.1.30.1      ?
v      \__   ttl    3                     ?/0           0 pps 192.1.30.2      192.1.30.2
Receiver      Query Source
```

Each line of the trace display is usually in the following format (depending on the options selected and the responses from the devices along the path):

```
hop-number host (ip-address) protocolttl
```

Table 105 on page 921 summarizes the output fields of the display.

NOTE: The packet statistics gathered from Juniper Networks devices and routing nodes always display as `0`.

Table 105: CLI mtrace from-source Command Output Summary

Field	Description
hop-number	Number of the hop (device) along the path.
host	Hostname, if available, or IP address of the device. If the `no-resolve` option was entered in the command, the hostname is not displayed.
ip-address	IP address of the device.
protocol	Protocol used.
ttl	TTL threshold.
`Round trip time` *milliseconds* `ms`	Total time between the sending of the query packet and the receiving of the response packet.
`total ttl of` *number* `required`	Total number of hops required to reach the source.
`Source`	Source IP address of the response packet.
`Response Dest`	Response destination IP address.
`Overall`	Average packet rate for all traffic at each hop.
`Packet Statistics For Traffic From`	Number of packets lost, number of packets sent, percentage of packets lost, and average packet rate at each hop.

Table 105: CLI mtrace from-source Command Output Summary *(Continued)*

Field	Description
`Receiver`	IP address receiving the multicast packets.
`Query Source`	IP address of the host sending the query packets.

Monitor Security Policies

SUMMARY

This section describes monitoring security policies and recording the permitted or denied traffic.

IN THIS SECTION

- Monitor Security Policy Statistics | 922

Monitor Security Policy Statistics

IN THIS SECTION

- Purpose | 922
- Action | 922

Purpose

Monitor and record traffic that Junos OS permits or denies based on previously configured policies.

Action

To monitor traffic, enable the count and log options.

Count—Configurable in an individual policy. If count is enabled, statistics are collected for sessions that enter the device for a given policy, and for the number of packets and bytes that pass through the device in both directions for a given policy. For counts (only for packets and bytes), you can specify that alarms be generated whenever the traffic exceeds specified thresholds. See count (Security Policies).

Log—Logging capability can be enabled with security policies during session initialization (**session-init**) or session close (**session-close**) stage. See log (Security Policies).

- To view logs from denied connections, enable log on **session-init**.
- To log sessions after their conclusion/tear-down, enable log on **session-close**.

> **NOTE**: Session log is enabled at real time in the flow code which impacts the user performance. If both **session-close** and **session-init** are enabled, performance is further degraded as compared to enabling **session-init** only.

For details about information collected for session logs, see Information Provided in Session Log Entries for SRX Series Services Gateways.

Monitor Interfaces and Switching Functions

SUMMARY

This section describes how to monitor interfaces and switching functions.

IN THIS SECTION

- Display Real-Time Interface Information | **924**
- Monitor Interfaces | **927**
- Monitor PPP | **928**

Display Real-Time Interface Information

Enter the `monitor interface` command to display real-time traffic, error, alarm, and filter statistics about a physical or logical interface:

```
user@host> monitor interface (interface-name | traffic)
```

Replace *interface-name* with the name of a physical or logical interface. If you specify the `traffic` option, statistics for all active interfaces display.

The real-time statistics update every second. The `Current delta` and `Delta` columns display the amount the statistics counters have changed since the `monitor interface` command was entered or since you cleared the delta counters. Table 106 on page 924 and Table 107 on page 925 list the keys you use to control the display using the *interface-name* and `traffic` options. (The keys are not case sensitive.)

Table 106: CLI monitor interface Output Control Keys

Key	Action
c	Clears (returns to 0) the delta counters in the `Current delta` column. The statistics counters are not cleared.
f	Freezes the display, halting the update of the statistics and delta counters.
i	Displays information about a different interface. You are prompted for the name of a specific interface.
n	Displays information about the next interface. The device scrolls through the physical and logical interfaces in the same order in which they are displayed by the `show interfaces terse` command.
q or ESC	Quits the command and returns to the command prompt.
t	Thaws the display, resuming the update of the statistics and delta counters.

Table 107: CLI monitor interface traffic Output Control Keys

Key	Action
b	Displays the statistics in units of bytes and bytes per second (bps).
c	Clears (returns to 0) the delta counters in the `Delta` column. The statistics counters are not cleared.
d	Displays the `Delta` column instead of the rate column—in bps or packets per second (pps).
p	Displays the statistics in units of packets and packets per second (pps).
q or ESC	Quits the command and returns to the command prompt.
r	Displays the rate column—in bps and pps—instead of the `Delta` column.

The following are sample displays from the `monitor interface` command:

```
user@host> monitor interface fe-0/0/0
```

```
host1                          Seconds: 5                  Time: 04:38:40
                                                         Delay: 3/0/10
Interface: fe-0/0/0, Enabled, Link is Up
Encapsulation: Ethernet, Speed: 1000mbps
Traffic statistics:                                        Current delta
  Input bytes:                885405423 (3248 bps)                [2631]
  Output bytes:               137411893 (3344 bps)               [10243]
  Input packets:                7155064 (2 pps)                     [28]
  Output packets:                636071 (1 pps)                     [23]
Error statistics:
  Input errors:                       0                             [0]
  Input drops:                        0                             [0]
  Input framing errors:               0                             [0]
  Policed discards:                   0                             [0]
  L3 incompletes:                     0                             [0]
```

```
  L2 channel errors:                  0                                [0]
  L2 mismatch timeouts:               0                                [0]
  Carrier transitions:                1                                [0]
  Output errors:                      0                                [0]
  Output drops:                       0                                [0]
  Aged packets:                       0                                [0]
Active alarms : None
Active defects: None
Input MAC/Filter statistics:
  Unicast packets                 73083                              [16]
  Broadcast packets             3629058                               [5]
  Multicast packets             3511364                               [3]
  Oversized frames                    0                                [0]
  Packet reject count                 0                                [0]
  DA rejects                          0                                [0]
  SA rejects                          0                                [0]
Output MAC/Filter Statistics:
  Unicast packets                629555                              [28]
  Broadcast packets                6494  Multicast packet              [0]
```

NOTE: The output fields that display when you enter the `monitor interface` *interface-name* command are determined by the interface you specify.

```
user@host> monitor interface traffic
```

```
Interface    Link  Input packets        (pps)     Output packets        (pps)  fe-0/0/0
Up          42334          (5)             23306          (3)  fe-0/0/1      Up
587525876      (12252)          589621478      (12891)
```

Monitor Interfaces

IN THIS SECTION

- Purpose | 927
- Action | 927

Purpose

View general information about all physical and logical interfaces for a device.

Action

Enter the following `show` commands in the CLI to view interface status and traffic statistics.

- `show interfaces terse`

  NOTE: On SRX Series Firewalls, when configuring identical IPs on a single interface, you will not see a warning message; instead, you will see a syslog message.

- `show interfaces extensive`
- `show interfaces` *`interface-name`*

NOTE: If you are using the J-Web user interfaces, select **Monitor>Interfaces** in the J-Web user interface. The J-Web Interfaces page displays the following details about each device interface:

- Port—Indicates the interface name.
- Admin Status—Indicates whether the interface is enabled (Up) or disabled (Down).
- Link Status—Indicates whether the interface is linked (Up) or not linked (Down).
- Address—Indicates the IP address of the interface.
- Zone—Indicates whether the zone is an untrust zone or a trust zone.
- Services—Indicates services that are enabled on the device, such as HTTP and SSH.

- Protocols—Indicates protocols that are enabled on the device, such as BGP and IGMP.
- Input Rate graph—Displays interface bandwidth utilization. Input rates are shown in bytes per second.
- Output Rate graph—Displays interface bandwidth utilization. Output rates are shown in bytes per second.
- Error Counters chart—Displays input and output error counters in the form of a bar chart.
- Packet Counters chart—Displays the number of broadcast, unicast, and multicast packet counters in the form of a pie chart. (Packet counter charts are supported only for interfaces that support MAC statistics.)

To change the interface display, use the following options:

- Port for FPC—Controls the member for which information is displayed.
- Start/Stop button—Starts or stops monitoring the selected interfaces.
- Show Graph—Displays input and output packet counters and error counters in the form of charts.
- Pop-up button—Displays the interface graphs in a separate pop-up window.
- Details—Displays extensive statistics about the selected interface, including its general status, traffic information, IP address, I/O errors, class-of-service data, and statistics.
- Refresh Interval—Indicates the duration of time after which you want the data on the page to be refreshed.
- Clear Statistics—Clears the statistics for the selected interface.

SEE ALSO

Interfaces User Guide for Security Devices

Monitor PPP

IN THIS SECTION

Purpose | 929

- Action | 929

Purpose

Display PPP monitoring information, including PPP address pool information, session status for PPP interfaces, cumulative statistics for all PPP interfaces, and a summary of PPP sessions.

NOTE: PPP monitoring information is available only in the CLI. The J-Web user interface does not include pages for displaying PPP monitoring information.

Action

Enter the following CLI commands:

- `show ppp address-pool` *`pool-name`*
- `show ppp interface` *`interface-name`*
- `show ppp statistics`
- `show ppp summary`

8

PART

Performance Management

Network Analytics | 931

Network Analytics

SUMMARY

This section describes the network analytics feature that provides visibility into the performance and behavior of the data center infrastructure. It collects data from the switch, analyzes the data by using sophisticated algorithms, and captures the results in reports. Network administrators can use the reports to troubleshoot problems, make decisions, and adjust resources as needed.

IN THIS SECTION

- Network Analytics Overview | **931**
- Understand Network Analytics Streaming Data | **942**
- Understand Enhanced Analytics Local File Output | **950**
- Understand Network Analytics Configuration and Status | **953**
- Configure Queue and Traffic Monitoring | **954**
- Configure a Local File for Network Analytics Data | **957**
- Configure a Remote Collector for Streaming Analytics Data | **958**
- Example: Configure Queue and traffic statsistics | **960**
- Example: Configure Queue and Traffic Monitoring | **969**

Network Analytics Overview

IN THIS SECTION

- Analytics Feature Overview | **932**
- Network Analytics Enhancements Overview | **933**
- Summary of CLI Changes | **935**

The analytics manager (analyticsm) in the Packet Forwarding Engine collects traffic and queue statistics, and the analytics daemon (analyticsd) in the Routing Engine analyzes the data and generates reports.

NOTE: In Junos OS Release 13.2X51-D15, the network analytics feature was enhanced, and extensive changes were made to the CLI statements and hierarchies. If you upgrade to Junos OS Release 13.2X51-D15 or later from a release prior to 13.2X51-D15, network analytics configurations committed in previous releases will appear on your device, but the feature is disabled. To enable this feature, you must reconfigure it by using the new CLI statements and hierarchies.

Analytics Feature Overview

You enable network analytics by configuring queue (microburst) monitoring and high-frequency traffic statistics monitoring.

Queue (microburst) monitoring:

You use microburst monitoring to look at traffic queue conditions on the network. A microburst occurrence indicates to the Packet Forwarding Engine that a user-specified queue depth or latency threshold is reached. The queue depth is the buffer (in bytes) containing the data, and latency is the time (in nanoseconds or microseconds) the data stays in the queue.

You can configure queue monitoring based on either queue depth or latency (but not both), and configure the frequency (polling interval) at which the Packet Forwarding Engine checks for microbursts and sends the data to the Routing Engine for processing. You may configure queue monitoring globally for all physical interfaces on the system, or for a specific interface on the switch. However, the specified queue monitoring interval applies either to all interfaces, or none; you cannot configure the interval for each interface.

High-frequency traffic statistics monitoring:

You use high-frequency traffic statistics monitoring to collect traffic statistics at specified polling intervals. Similar to the queue monitoring interval, the traffic monitoring interval applies either to all interfaces, or none; you cannot configure the interval for each interface.

Both traffic and queue monitoring are disabled by default. You must configure each type of monitoring using the CLI. In each case, the configuration for an interface always takes precedence over the global configuration.

NOTE: You can configure traffic and queue monitoring for physical interfaces only; logical interfaces and Virtual Chassis port (VCP) interfaces are not supported.

The analyticsd daemon in the Routing Engine generates local log files containing queue and traffic statistics records. You can specify the log filename and size, and the number of log files. If you do not configure a filename, the data is not saved.

You can display the local log file or specify a server to receive the streaming data containing the queue and traffic statistics.

For each port, information for the last 10 records of traffic statistics and 100 records of queue statistics is cached. You may view this information by using the `show analytics` commands.

To store traceoptions data, you configure the `traceoptions` statement at the `[edit services analytics]` hierarchy level.

Network Analytics Enhancements Overview

Beginning in Junos OS Release 13.2X51-D15, the network analytics feature provides the following enhancements:

- Resources—Consist of interfaces and system. The interfaces resource allows you to configure an interface name and an associated resource profile name for each interface. With the system resource, you can configure the polling intervals for queue monitoring and traffic monitoring, and an associated resource profile for the system.
- Resource profile—A template that contains the configurations for queue and traffic monitoring, such as depth threshold and latency threshold values, and whether each type of monitoring is enabled or disabled. Once a resource profile is configured, you apply it to a system or interfaces resource.
- Collector—A server for collecting queue and traffic monitoring statistics, and can be a local or remote server. You can configure a local server to store monitoring statistics in a log file, or a remote server to receive streamed statistics data.
- Export profile—You must configure an export profile if you wish to send streaming data to a remote collector. In the export profile, you define the category of streamed data (system-wide or interface-specific) to determine stream type the collector will receive. You can specify both system and interface stream categories. System data includes system information and status of queue and traffic monitoring. Interface-specific data includes interface information, queue and traffic statistics, and link, queue, and traffic status.
- Google Protocol Buffer (GBP) stream format—A new streaming format for monitoring statistics data that is sent to a remote collector in a single AnRecord message. The format of this stream which provides nine types of information is shown in Table 108 on page 934.

Table 108: Google Protocol Buffer (GBP) stream format

Message	Description
System information	General system information, including boot time, model information, serial number, number of ports, and so on
System queue status	Queue status for the system in general
System traffic status	Traffic status for the system in general
Interface information	Includes SNMP index, slot, port, and other information
Queue statistics for interfaces	Queue statistics for specific interfaces
Traffic statistics for interfaces	Traffic statistics for specific interfaces
Link status for interfaces	Includes link speed, state, and so on
Queue status for interfaces	Queue status for specific interfaces
Traffic status for interfaces	Traffic status for specific interfaces

- The analytics.proto file—Provides a template for the GBP stream format. This file can be used for writing your analytics server application. To download the file, go to: /documentation/en_US/junos13.2/topics/reference/proto-files/analytics-proto.txt
- Use of threshold values—The Analytics Manager (analyticsm) will generate a queue statistics record when the lower queue depth or latency threshold value is exceeded.
- User Datagram Protocol (UDP)—Additional transport protocol you can configure, in addition to Transmission Control Protocol (TCP), for the remote streaming server port.
- Single file for local logging—Replaces the separate log files for queue and traffic statistics.
- Change in latency measurement—Configuration and reporting of latency values have changed from microseconds to nanoseconds.

- Change in reporting of the collection time in UTC format—Statistics collection time is reported in microseconds instead of milliseconds.
- New operational mode command `show analytics collector`—Replaces the `show analytics streaming-server` command.
- Changes in command output format—Include the following changes:
  - Addition of unicast, multicast, and broadcast packet counters in queue and traffic statistics.
  - Reversal of the sequence of statistics information in the output. The most recent record is displayed at the beginning, and the oldest record at the end of the output.
  - Removal of traffic or queue monitoring status information from the global portion of the `show analytics configuration` and `show analytics status` command output if there is no global configuration.
  - Addition of `n/a` to the interface-specific portion of the `show analytics configuration` and `show analytics status` command output if a parameter is not configured (for example, depth threshold or latency threshold).

Summary of CLI Changes

Starting in Junos OS Release 13.2X51-D15, enhancements to the network analytics feature result in changes in the CLI when you configure the feature. See Table 109 on page 935 for a summary of CLI changes.

Table 109: Network Analytics CLI Changes

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later

Configuring global queue and traffic monitoring polling interval	`[edit services analytics]`
	
	`traffic-statistics {`
	`  interval interval;`
	`}`
	`queue-statistics {`
	`  interval interval;`
	`}`
		`[edit services analytics]`
		
		`resource {`
		`  system {`
		`  polling-interval {`
		`  queue-monitoring interval;`
		`  traffic-monitoring interval;`
		`  }`
		`  }`
		`}`

Table 109: Network Analytics CLI Changes *(Continued)*

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later

Configuring local files for traffic and queue statistics reporting	[edit services analytics]
	
	traffic-statistics {
	file *filename*;
	size *size*;
	files *number*;
	}
	queue-statistics {
	file *filename*;
	size *size*;
	files *number*;
	}
		[edit services analytics]
		
		collector {
		local {
		file *filename* {
		files *number*;
		size *size*;
		}
		}
		}

Table 109: Network Analytics CLI Changes *(Continued)*

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later

Enabling queue statistics and traffic monitoring, and specifying the depth threshold for all interfaces (globally)	`[edit services analytics]`
	
	`interfaces {`
	`  all {`
	`  queue-statistics;`
	`  traffic-statistics;`
	`  depth-threshold {`
	`  high *number*;`
	`  low *number*;`
	`  }`
	`  }`
	`}`
		Requires defining a resource profile and applying it to the system:
		
		1. To define a resource profile:
		
		`[edit services analytics]`
		
		`resource-profiles {`
		`  *profile-name*{`
		`  queue-monitoring;`
		`  traffic-monitoring;`
		`  depth-threshold {`
		`  high *number*;`
		`  low *number*;`
		`  }`
		`  }`
		`}`
		
		2. To apply a profile to the system:
		
		`[edit services analytics]`
		
		`resource {`
		`  system {`
		`  resource-profile *profile-name*;`
		`  }`
		`}`

Table 109: Network Analytics CLI Changes *(Continued)*

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later

Enabling queue statistics and traffic monitoring, and specifying the latency threshold for one interface	`[edit services analytics]`
	
	`interfaces {`
	`  *interface*{`
	`  queue-statistics;`
	`  traffic-statistics;`
	`  latency-threshold`
	`  high *number*;`
	`  low *number*;`
	`  }`
	`}`
		Requires defining a resource profile and applying it to the interface:
		
		1. To define a resource profile:
		
		`[edit services analytics]`
		
		`resource-profiles {`
		`  *profile-name*{`
		`  queue-monitoring;`
		`  traffic-monitoring;`
		`  latency-threshold {`
		`  high *number*;`
		`  low *number*;`
		`  }`
		`  }`
		`}`
		
		2. To apply a profile to the interface:
		
		`[edit services analytics]`
		
		`resource {`
		`  interfaces {`
		`  *interface-name* {`
		`  resource-profile *profile-name*;`
		`  }`
		`  }`
		`}`

Table 109: Network Analytics CLI Changes *(Continued)*

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later

Configuring the streaming data format (JSON, CSV, or TSV) to send to a remote server
 	
NOTE: Junos OS Release 13.2X51-D15 added support for the GPB stream format and configuration of the transport protocols (TCP or UDP).
	`[edit services analytics]`
	
	`streaming-servers {`
	`  address ip-address {`
	`  port number {`
	`  stream-format format;`
	`  }`
	`  }`
	`}`
		Requires defining the stream format in an export profile and applying the profile to the collector.
		
		1. To configure the stream format:
		
		`[edit services analytics]`
		
		`export-profiles {`
		`  profile-name {`
		`  stream-format format;`
		`  }`
		`}`
		
		2. To apply an export profile to the collector:
		
		`[edit services analytics]`
		
		`collector {`
		`  address ip-address {`
		`  port number {`
		`  transport protocol {`
		`  export-profile profile-name;`
		`  }`
		`  }`
		`  }`
		`}`

Table 109: Network Analytics CLI Changes *(Continued)*

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later

Configuring the streaming message types (queue or traffic statistics) to send to a remote server	`[edit services analytics]`
	
	`streaming-servers {`
	`  address ip-address {`
	`  port number {`
	`  stream-type type;`
	`  stream-type type;`
	`  }`
	`  }`
	`}`
		Requires defining an export profile and applying it to the collector:
		
		1. To define an export profile:
		
		`[edit services analytics]`
		
		`export-profiles {`
		`  profile-name {`
		`  interface {`
		`  information;`
		`  statistics {`
		`  queue;`
		`  traffic;`
		`  }`
		`  status {`
		`  link;`
		`  queue;`
		`  traffic;`
		`  }`
		`  }`
		`  system {`
		`  information;`
		`  status {`
		`  queue;`
		`  traffic;`
		`  }`
		`  }`
		`  }`
		`}`
		
		2. To apply an export profile to the collector:
		
		`[edit services analytics]`
		
		`collector {`
		`  address ip-address {`
		`  port number {`

Table 109: Network Analytics CLI Changes *(Continued)*

Task	CLI for Junos OS Release 13.2X50-D15 and 13.2X51-D10	CLI for Junos OS Release 13.2X51-D15 and later
		export-profile *profile-name*; } } }

Configuring the transport protocol for sending streaming data to an external server	No configuration is available. Only the TCP protocol is supported.	Configuration is available. Both TCP and UDP protocols are supported, and can be configured for the same port.
		[edit services analytics]
		collector {
		address *ip-address* {
		port *number1* {
		transport tcp;
		transport udp;
		}
		port *number2* {
		transport udp;
		}
		}
		}

Show information about remote streaming server or collector	Issue the show analytics streaming-sever command.	Issue the show analytics collector command.

Understand Network Analytics Streaming Data

IN THIS SECTION

- JavaScript Object Notation (JSON) | 942
- Comma-separated Values (CSV) | 943
- Tab-separated Values (TSV) | 943
- Google Protocol Buffer (GPB) | 946

Network analytics monitoring data can be streamed to remote servers called collectors. You can configure one or more collectors to receive streamed data containing queue and traffic statistics. This topic describes the streamed data output.

In Junos OS Release 13.2X51-D10, network analytics provide support for the following streaming data formats and output:

- JavaScript Object Notation (JSON)
- Comma-separated Values (CSV)
- Tab-separated Values (TSV)

NOTE: For the output shown in this topic for JSON, CSV, and TSV formats, the time is displayed in the Unix epoch format (also known as Unix time or POSIX time).

Starting in Junos OS Release 13.2X51-D15, support for the below streaming format and output has been added along with JSON, CSV, and TSV formats.

- Google Protocol Buffer (GPB)

JavaScript Object Notation (JSON)

The JavaScript Object Notation (JSON) streaming format supports the following data:

- Queue statistics data. For example:

```
{"record-type":"queue-stats","time":1383453988263,"router-id":"qfx5100-switch",
"port":"xe-0/0/18","latency":0,"queue-depth":208}
```

See Table 110 on page 944 for more information about queue statistics output fields.

- Traffic statistics. For example:

```
{"record-type":"traffic-stats","time":1383453986763,"router-id":"qfx5100-switch",
"port":"xe-0/0/16","rxpkt":26524223621,"rxpps":8399588,"rxbyte":3395100629632,
"rxbps":423997832,"rxdrop":0,"rxerr":0,"txpkt":795746503,"txpps":0,"txbyte":101855533467,
"txbps":0,"txdrop":0,"txerr":0}
```

See Table 111 on page 945 for more information about traffic statistics output fields.

Comma-separated Values (CSV)

The Comma-separated Values (CSV) streaming format supports the following data:

- Queue statistics. For example:

```
q,1383454067604,qfx5100-switch,xe-0/0/18,0,208
```

See Table 110 on page 944 for more information about queue statistics output fields.

- Traffic statistics. For example:

```
t,1383454072924,qfx5100-switch,xe-0/0/19,1274299748,82950,163110341556,85603312,0,0,
27254178291,8300088,3488534810679,600002408,27268587050,3490379142400
```

See Table 111 on page 945 for more information about traffic statistics output fields.

Tab-separated Values (TSV)

The Tab-separated Values (TSV) streaming format supports the following data:

- Queue statistics. For example:

```
q       585870192561703872      qfx5100-switch       xe-0/0/18       (null)
208     2
```

See Table 110 on page 944 for more information about queue statistics output fields.

- Traffic statistics. For example:

```
t       1383454139025   qfx5100-switch  xe-0/0/19       1279874033      82022
163823850036    84801488        0       0       27811618258     8199630
3559887126455   919998736       27827356915     3561901685120
```

See Table 111 on page 945 for more information about traffic statistics output fields.

Queue Statistics Output for JSON, CSV, and TSV

Table 110 on page 944 describes the output fields for streamed queue statistics data in the order they appear.

Table 110: Streamed Queue Statistics Data Output Fields

Field	Description
record-type	Type of statistics. Displayed as: • `queue-stats` (JSON format) • `q` (CSV or TSV format)
time	Time (in Unix epoch format) at which the statistics were captured.
router-id	ID of the network analytics host device.
port	Name of the physical port configured for network analytics.
latency	Traffic queue latency in milliseconds.

Table 110: Streamed Queue Statistics Data Output Fields *(Continued)*

Field	Description
queue depth	Depth of the traffic queue in bytes.

Traffic Statistics Output for JSON, CSV, and TSV

Table 111 on page 945 describes the output fields for streamed traffic statistics data in the order they appear.

Table 111: Streamed Traffic Statistics Data Output Fields

Field	Description
record-type	Type of statistics. Displayed as: • `traffic-stats` (JSON format) • t (CSV or TSV format)
time	Time (in Unix epoch format) at which the statistics were captured.
router-id	ID of the network analytics host device.
port	Name of the physical port configured for network analytics.
rxpkt	Total packets received.
rxpps	Total packets received per second.
rxbyte	Total bytes received.
rxbps	Total bytes received per second.
rxdrop	Total incoming packets dropped.

Table 111: Streamed Traffic Statistics Data Output Fields *(Continued)*

Field	Description
rxerr	Total packets with errors.
txpkt	Total packets transmitted.
txpps	Total packets transmitted per second.
txbyte	Total bytes transmitted.
txbps	Total bytes transmitted per second.
txdrop	Total transmitted bytes dropped.
txerr	Total transmitted packets with errors (dropped).

Google Protocol Buffer (GPB)

This streaming format provides:

- Support for nine types of messages, based on resource type (system-wide or interface-specific).
- Sends messages in a hierarchical format.
- You can generate other stream format messages (JSON, CSV, TSV) from GPB formatted messages.
- Includes a 8-byte message header. See Table 112 on page 946 for more information.

Table 112 on page 946 describes the GPB stream format message header.

Table 112: GPB Stream Format Message Header Information

Byte Position	Field
0 to 3	Length of message

Table 112: GPB Stream Format Message Header Information *(Continued)*

Byte Position	Field
4	Message version
5 to 7	Reserved for future use

The following GPB prototype file (**analytics.proto**) provides details about the streamed data:

```
package analytics;

// Traffic statistics related info
message TrafficStatus {
    optional uint32             status          = 1;
    optional uint32             poll_interval   = 2;
}

// Queue statistics related info
message QueueStatus {
    optional uint32             status          = 1;
    optional uint32             poll_interval   = 2;
    optional uint64             lt_high         = 3;
    optional uint64             lt_low          = 4;
    optional uint64             dt_high         = 5;
    optional uint64             dt_low          = 6;
}

message LinkStatus {
    optional uint64             speed           = 1;
    optional uint32             duplex          = 2;
    optional uint32             mtu             = 3;
    optional bool               state           = 4;
    optional bool               auto_negotiation= 5;
}

message InterfaceInfo {
    optional uint32             snmp_index      = 1;
    optional uint32             index           = 2;
    optional uint32             slot            = 3;
```

```
    optional uint32             port            = 4;
    optional uint32             media_type      = 5;
    optional uint32             capability      = 6;
    optional uint32             porttype        = 7;
}

message InterfaceStatus {
    optional LinkStatus         link            = 1;
    optional QueueStatus        queue_status    = 2;
    optional TrafficStatus      traffic_status  = 3;
}

message QueueStats {
    optional uint64             timestamp       = 1;
    optional uint64             queue_depth     = 2;
    optional uint64             latency         = 3;
}

message TrafficStats {
    optional uint64             timestamp       = 1;
    optional uint64             rxpkt           = 2;
    optional uint64             rxucpkt         = 3;
    optional uint64             rxmcpkt         = 4;
    optional uint64             rxbcpkt         = 5;
    optional uint64             rxpps           = 6;
    optional uint64             rxbyte          = 7;
    optional uint64             rxbps           = 8;
    optional uint64             rxcrcerr        = 9;
    optional uint64             rxdroppkt       = 10;
    optional uint64             txpkt           = 11;
    optional uint64             txucpkt         = 12;
    optional uint64             txmcpkt         = 13;
    optional uint64             txbcpkt         = 14;
    optional uint64             txpps           = 15;
    optional uint64             txbyte          = 16;
    optional uint64             txbps           = 17;
    optional uint64             txcrcerr        = 18;
    optional uint64             txdroppkt       = 19;
}

message InterfaceStats {
    optional TrafficStats       traffic_stats   = 1;
    optional QueueStats         queue_stats     = 2;
```

```
}

//Interface message
message Interface {
    required string             name            = 1;
    optional bool               deleted         = 2;
    optional InterfaceInfo      information     = 3;
    optional InterfaceStats     stats           = 4;
    optional InterfaceStatus    status          = 5;
}

message SystemInfo {
    optional uint64             boot_time       = 1;
    optional string             model_info      = 2;
    optional string             serial_no       = 3;
    optional uint32             max_ports       = 4;
    optional string             collector       = 5;
    repeated string             interface_list  = 6;
}

message SystemStatus {
    optional QueueStatus        queue_status    = 1;
    optional TrafficStatus      traffic_status  = 2;
}

//System message
message System {
    required string             name            = 1;
    optional bool               deleted         = 2;
    optional SystemInfo         information     = 3;
    optional SystemStatus       status          = 4;
}

message AnRecord {
    optional uint64             timestamp       = 1;
    optional System             system          = 2;
    repeated Interface          interface       = 3;
}
```

SEE ALSO

collector (Analytics)

Understand Enhanced Analytics Local File Output

The network analytics feature provides visibility into the performance and behavior of the data center infrastructure. You enable network analytics by configuring queue or traffic statistics monitoring, or both. In addition, you can configure a local file for storing the traffic and queue statistics records.

NOTE: This topic describes the local file output in Junos OS Release 13.2X51-D15 and later.

Starting in Junos OS Release 13.2X51-D15, the traffic and queue monitoring statistics can be stored locally in a single file. The following example shows the output from the `monitor start` command.

```
root@qfx5100-33> monitor start an

root@qfx5100-33>
*** an ***
q,1393947567698432,qfx5100-33,xe-0/0/19,1098572,1373216
q,1393947568702418,qfx5100-33,xe-0/0/19,1094912,1368640
q,1393947569703415,qfx5100-33,xe-0/0/19,1103065,1378832
t,1393947569874528,qfx5100-33,xe-0/0/16,12603371884,12603371884,0,0,
8426023,1613231610488,8628248712,0,3,5916761,5916761,0,0,0,757345408,0,0,0
t,1393947569874528,qfx5100-33,xe-0/0/18,12601953614,12601953614,0,0,
8446737,1613050071660,8649421552,0,5,131761619,131761619,0,0,84468,
16865487232,86495888,0,0
t,1393947569874528,qfx5100-33,xe-0/0/19,126009250,126009250,0,0,84469,
16129184128,86496392,0,0,12584980342,12584980342,0,0,8446866,1610877487744,
8649588432,12593703960,0
q,1393947575698402,qfx5100-33,xe-0/0/19,1102233,1377792
q,1393947576701398,qfx5100-33,xe-0/0/19,1107724,1384656
```

See Table 113 on page 951 for queue statistics output, and Table 114 on page 951 for traffic statistics output. The fields in the tables are listed in the order they appear in the output example.

Table 113: Output Fields for Queue Statistics in Local Analytics File

Field	Description	Example in Output
Record type	Type of statistics (queue or traffic monitoring)	`q`
Time (microseconds)	Unix epoch (or Unix time) in microseconds at which the statistics were captured.	`1393947567698432`
Router ID	ID of the network analytics host device.	`qfx5100-33`
Port	Name of the physical port configured for network analytics.	`xe-0/0/19`
Latency (nanoseconds)	Traffic queue latency in nanoseconds.	`1098572`
Queue depth (bytes)	Depth of the traffic queue in bytes.	`1373216`

Table 114: Output Fields for Traffic Statistics in Local Analytics File

Field	Description	Example in Output
Record type	Type of statistics (queue or traffic monitoring)	`t`
Time (microseconds)	Unix epoch (or Unix time) in microseconds at which the statistics were captured.	`1393947569874528`
Router ID	ID of the network analytics host device.	`qfx5100-33`
Port	Name of the physical port configured for network analytics.	`xe-0/0/16`
rxpkt	Total packets received.	`12603371884`
rxucpkt	Total unicast packets received.	`12603371884`
rxmcpkt	Total multicast packets received.	`0`

Table 114: Output Fields for Traffic Statistics in Local Analytics File *(Continued)*

Field	Description	Example in Output
rxbcpkt	Total broadcast packets received.	0
rxpps	Total packets received per second.	8426023
rxbyte	Total octets received.	1613231610488
rxbps	Total bytes received per second.	8628248712
rxdroppkt	Total incoming packets dropped.	0
rxcrcerr	CRC/Align errors received.	3
txpkt	Total packets transmitted.	5916761
txucpkt	Total unicast packets transmitted.	5916761
txmcpkt	Total multicast packets transmitted.	0
txbcpkt	Total broadcast packets transmitted.	0
txpps	Total packets transmitted per second.	0
txbyte	Total octets transmitted.	757345408
txbps	Bytes per second transmitted.	0
txdroppkt	Total transmitted packets dropped.	0
txcrcerr	CRC/Align errors transmitted.	0

Understand Network Analytics Configuration and Status

The network analytics feature provides visibility into the performance and behavior of the data center infrastructure. You can enable network analytics by configuring traffic and queue statistics monitoring.

> **NOTE**: This topic describes the configuration and status output from Junos OS Release 13.2X50-D15 and 13.2X51-D10 only.

If you had enabled traffic or queue monitoring, you can issue the `show analytics configuration` and `show analytics status` commands to view the global interface configuration and status and that of specific interfaces. The output that is displayed depends on your configuration at the global interface and specific interface levels. For example:

- A global interface configuration (for all interfaces) to disable monitoring supersedes the configuration to enable it on an interface.
- The interface configuration to enable or disable monitoring supersedes the global interface configuration, unless monitoring had been disabled globally for all interfaces.
- If there is no configuration, whether for all interfaces or a specific interface, monitoring is disabled by default (see Table 115 on page 953).

Table 115 on page 953 describes the correlation between the user configuration and the settings that are displayed.

Table 115: Configuration and Status Output in Junos OS Release 13.2X51-D10 and 13.2X50-D15

User Configuration	Global or System Settings		Specific Interface Settings	
	Configuration	Status	Configuration	Status
No global or specific interface configuration. This is the default setting.	Auto	Auto	Auto	Disabled
No global interface configuration but the specific interface monitoring is disabled.	Auto	Auto	Disabled	Disabled
No global interface configuration but the specific interface monitoring is enabled.	Auto	Auto	Enabled	Enabled

Table 115: Configuration and Status Output in Junos OS Release 13.2X51-D10 and 13.2X50-D15 ***(Continued)***

User Configuration	Global or System Settings		Specific Interface Settings	
	Configuration	Status	Configuration	Status
Monitoring is disabled globally and there is no interface configuration.	Disabled	Disabled	Auto	Disabled
Monitoring is disabled at both the global and specific interface levels.	Disabled	Disabled	Disabled	Disabled
Monitoring is disabled at the global interface level but is enabled at the specific interface level. The global interface *Disabled* setting supersedes the *Enabled* setting for a specific interface.	Disabled	Disabled	Enabled	Disabled
Monitoring is enabled for all interfaces but there is no configuration for the specific interface .	Enabled	Enabled	Auto	Enabled
Monitoring is enabled at both the global and specific interface levels.	Enabled	Enabled	Enabled	Enabled
Monitoring is enabled for all interfaces but is disabled for the specific interface.	Enabled	Enabled	Disabled	Disabled

SEE ALSO

queue-statistics

traffic-statistics

Configure Queue and Traffic Monitoring

Network analytics queue and traffic monitoring provides visibility into the performance and behavior of the data center infrastructure. This feature collects data from the switch, analyzes the data using

sophisticated algorithms, and captures the results in reports. You can use the reports to help troubleshoot problems, make decisions, and adjust resources as needed.

You enable queue and traffic monitoring by first defining a resource profile template, and then applying the profile to the system (for a global configuration) or to individual interfaces.

NOTE: You can configure queue and traffic monitoring on physical network interfaces only; logical interfaces and Virtual Chassis physical (VCP) interfaces are not supported.

NOTE: The procedure to configure queue and traffic monitoring on a QFX Series standalone switch requires Junos OS Release 13.2X51-D15 or later to be installed on your device.

To configure queue monitoring on a QFX Series standalone switch:

1. Configure the queue monitoring polling interval (in milliseconds) globally (for the system):

```
[edit]
set services analytics resource system polling-interval queue-monitoring interval
```

2. Configure a resource profile for the system, and enable queue monitoring:

```
[edit]
set services analytics resource-profiles profile-name queue-monitoring
```

3. Configure high and low values of the depth-threshold (in bytes) for queue monitoring in the system profile:

```
[edit]
set services analytics resource-profiles profile-name depth-threshold high number low number
```

For both high and low values, the range is from 1 to 1,250,000,000 bytes, and the default value is 0 bytes.

NOTE: You can configure either the depth-threshold or latency threshold for the system, but not both.

4. Apply the resource profile template to the system for a global configuration:

```
[edit]
set services analytics resource system resource-profile profile-name
```

5. Configure an interface-specific resource profile and enable queue monitoring for the interface:

```
[edit]
set services analytics resource-profiles profile-name queue-monitoring
```

6. Configure the latency-threshold (high and low values) for queue monitoring in the interface-specific profile:

```
[edit]
set services analytics resource-profiles profile-name latency-threshold high number low number
```

   For both high and low values, the range is from 1 to 100,000,000 nanoseconds, and the default value is 1,000,000 nanoseconds.

   NOTE: You can configure either the depth-threshold or latency threshold for interfaces, but not both.

7. Apply the resource profile template for interfaces to one or more interfaces:

```
[edit]
set services analytics resource interfaces interface-name resource-profile profile-name
```

   NOTE: If a conflict arises between the system and interface configurations, the interface-specific configuration supersedes the global (system) configuration.

To configure traffic monitoring on a QFX Series standalone switch:

1. Configure the traffic monitoring polling interval (in seconds) for the system:

```
[edit]
set services analytics resource system polling-interval traffic-monitoring interval
```

2. Configure a resource profile for the system, and enable traffic monitoring in the profile:

```
[edit]
set services analytics resource-profiles profile-name traffic-monitoring
```

3. Apply the resource profile to the system for a global configuration:

```
[edit]
set services analytics resource system resource-profile profile-name
```

4. Configure a resource profile for interfaces, and enable traffic monitoring in the profile:

```
[edit]
set services analytics resource-profiles profile-name traffic-monitoring
```

> **NOTE**: If a conflict arises between the system and interface configurations, the interface-specific configuration supersedes the global (system) configuration.

5. Apply the resource profile template to one or more interfaces:

```
[edit]
set services analytics resource interfaces interface-name resource-profile profile-name
```

Configure a Local File for Network Analytics Data

The network analytics feature provides visibility into the performance and behavior of the data center infrastructure. This feature collects data from the switch, analyzes the data using sophisticated algorithms, and captures the results in reports. Network administrators can use the reports to help troubleshoot problems, make decisions, and adjust resources as needed.

To save the queue and traffic statistics data in a local file, you must configure a filename to store it.

NOTE: The procedure to configure a local file for storing queue and traffic monitoring statistics requires Junos OS Release 13.2X51-D15 or later to be installed on your device.

To configure a local file for storing queue and traffic monitoring statistics:

1. Configure a filename:

```
[edit]
set services analytics collector local file filename
```

   There is no default filename. If you do not configure a filename, network analytics statistics are not saved locally.

2. Configure the number of files (from 2 to 1000 files):

```
[edit]
set services analytics collector local file filename files number
```

3. Configure the file size (from 10 to 4095 MB) in the format of *x*m:

```
[edit]
set services analytics collector local file an size size
```

Configure a Remote Collector for Streaming Analytics Data

The network analytics feature provides visibility into the performance and behavior of the data center infrastructure. This feature collects data from the switch, analyzes the data using sophisticated algorithms, and captures the results in reports. Network administrators can use the reports to help troubleshoot problems, make decisions, and adjust resources as needed.

You can configure an export profile to define the stream format and type of data, and one or more remote servers (collectors) to receive streaming network analytics data.

NOTE: The procedure to configure a collector for receiving streamed analytics data requires Junos OS Release 13.2X51-D15 or later to be installed on your device.

To configure a collector for receiving streamed analytics data:

1. Create an export profile and specify the stream format:

```
[edit]
set services analytics export-profiles profile-name stream-format format
```

2. Configure the export profile to include interface information:

```
[edit]
set services analytics export-profiles profile-name interface information
```

3. Configure the export profile to include interface queue statistics:

```
[edit]
set services analytics export-profiles profile-name interface statistics queue
```

4. Configure the export profile to include interface traffic statistics:

```
[edit]
set services analytics export-profiles profile-name interface statistics traffic
```

5. Configure the export profile to include interface status link information:

```
[edit]
set services analytics export-profiles profile-name interface status link
```

6. Configure the export profile to include system information:

```
[edit]
set services analytics export-profiles profile-name system information
```

7. Configure the export profile to include system queue status:

```
[edit]
set services analytics export-profiles profile-name system status queue
```

8. Configure the export profile to include system traffic status:

```
[edit]
set services analytics export-profiles profile-name system status traffic
```

9. Configure the transport protocol for the collector addresses and apply the export profile:

```
[edit]
set services analytics collector address ip-address port port transport protocol export-
profile profile-name
set services analytics collector address ip-address port port transport protocol export-
profile profile-name
```

NOTE: If you configure the `tcp` or `udp` option for the JSON, CSV, and TSV formats, you must also set up the TCP or UDP client software on the remote collector to process records that are separated by the newline character (\n) on the remote server.

If you configure the `tcp` or `udp` option for the GPB format, you must also set up the TCP or UDP build streaming server using the **analytics.proto** file.

Example: Configure Queue and traffic statsistics

IN THIS SECTION

- Requirements | **961**
- Overview | **961**
- Configuration | **962**
- Verification | **965**

This example shows how to configure network analytics which includes queue and traffic monitoring on a QFX3500 standalone switch.

NOTE: The configuration shown in this example is supported only on Junos OS Release 13.2X50-D15 and 13.2X51-D10.

Requirements

This example uses the following hardware and software components:

- A QFX3500 standalone switch
- A external streaming server to collect data
- Junos OS Release 13.2X50-D15 software
- TCP server software (for remote streaming servers)

Before you configure network analytics, be sure you have:

- Junos OS Release 13.2X50-D15 or later software installed and running on the QFX3500 switch
- (Optional for streaming servers) TCP server software set up for processing records separated by a newline character (\n) on the remote streaming server
- All other devices running

Overview

IN THIS SECTION

Topology | 962

The network analytics feature provides visibility into the performance and behavior of the data center infrastructure. This feature collects data from the switch, analyzes the data using sophisticated algorithms, and captures the results in reports. Network administrators can use the reports to help troubleshoot problems, make decisions, and adjust resources as needed. You can enable network analytics by configuring queue and traffic statistics monitoring.

Topology

In this example, the QFX3500 switch is connected to an external server used for streaming statistics data.

Configuration

IN THIS SECTION

- CLI Quick Configuration | **962**
- Configure Queue and Traffic Statistics Monitoring | **963**
- Configure Local Statistics Files | **963**
- Configure Streaming Servers | **964**
- Results | **965**

To configure network analytics, perform these tasks:

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `[edit]` hierarchy level.

```
[edit]
set services analytics interfaces all queue-statistics
set services analytics interfaces all latency-threshold high 900 low 300
set services analytics interfaces xe-0/0/1 traffic-statistics
set services analytics queue-statistics file qstats1.qs files 3 size 10
set services analytics queue-statistics interval 10
set services analytics traffic-statistics file tstats1.ts files 3 size 10
set services analytics traffic-statistics interval 2
set services analytics streaming-servers address 10.94.198.11 port 50001 stream-format json
stream-type queue-statistics
set services analytics streaming-servers address 10.94.198.11 port 50005 stream-format csv
stream-type traffic-statistics
```

Configure Queue and Traffic Statistics Monitoring

Step-by-Step Procedure

To configure queue and traffic monitoring on physical interfaces:

> **NOTE**: Disabling of the queue or traffic monitoring supersedes the configuration (enabling) of this feature. You disable monitoring by issuing the `no-queue-statistics` or `no-traffic-statistics` at the `[edit services analytics interfaces]` hierarchy level.

1. Configure all interfaces for queue monitoring and set the latency thresholds (in microseconds):

```
[edit]
set services analytics interfaces all queue-statistics
set services analytics interfaces all latency-threshold high 900 low 300
```

2. Configure one interface for traffic monitoring:

```
[edit]
set services analytics interfaces xe-0/0/1 traffic-statistics
```

Configure Local Statistics Files

Step-by-Step Procedure

To configure local statistics files:

1. Configure the number of queue statistics files, and each file size in MB:

```
[edit]
set services analytics queue-statistics file qstats1.qs files 3 size 10m
```

2. Configure the queue statistics collection interval in milliseconds

```
[edit]
set services analytics queue-statistics interval 10
```

3. Configure the number of traffic statistics files, and each file size in MB:

```
[edit]
set services analytics traffic-statistics file tstats1.ts files 3 size 10m
```

4. Configure the traffic statistics collection interval in seconds:

```
[edit]
set services analytics traffic-statistics interval 2
```

Configure Streaming Servers

Step-by-Step Procedure

To configure streaming servers for receiving monitoring data:

> **NOTE**: In addition to configuring streaming servers, you must also set up the TCP client software to process records that are separated by the newline character (\n) on the remote server.

1. Configure a server IP address and port for queue statistics monitoring:

```
[edit]
set services analytics streaming-servers address 10.94.198.11 port 50001 stream-format json
stream-type queue-statistics
```

2. Configure a server IP address and port for traffic statistics monitoring:

```
[edit]
set services analytics streaming-servers address 10.94.198.11 port 50005 stream-format csv
stream-type traffic-statistics
```

Results

Display the results of the configuration:

```
[edit services analytics]
user@switch> show configuration
    queue-statistics {
        file qstats1.qs size 10m files 3;
        interval 10;
    }
    traffic-statistics {
        file tstats1.ts size 10m files 3;
        interval 2;
    }
    interfaces {
        xe-0/0/1 {
            traffic-statistics;
        }
        all {
            queue-statistics;
            latency-threshold high 900 low 300;
        }
    }
```

Verification

IN THIS SECTION

- Verify the Network Analytics Configuration | **966**
- Verify the Network Analytics Status | **966**
- Verify Streaming Servers Configuration | **967**
- Verify Queue Statistics | **967**
- Verify Traffic Statistics | **968**

Confirm that the configuration is correct and works as expected by performing these tasks:

Verify the Network Analytics Configuration

Purpose

Verify the configuration for network analytics.

Action

From operational mode, enter the `show analytics configuration` command to display the traffic and queue monitoring configuration.

```
user@host> show analytics configuration

Global configurations:
   Traffic statistics: Auto, Poll interval: 2 seconds
   Queue statistics: Enabled, Poll interval: 10 milliseconds
   Depth threshold high: 0 bytes, low: 0 bytes
   Latency threshold high: 900 microseconds, low: 300 microseconds
Interface     Traffic       Queue        Depth-threshold      Latency-threshold
            Statistics    Statistics      High      Low         High      Low
                                             (bytes)           (microseconds)
xe-0/0/1     Enabled      Auto           0         0          900       300
```

Meaning

The output displays information about traffic and queue monitoring on the switch.

Verify the Network Analytics Status

Purpose

Verify the network analytics operational status of the switch.

Action

From operational mode, enter the `show analytics status` command to display the traffic and queue monitoring status.

```
user@host> show analytics status
```

```
Global configurations:
   Traffic statistics: Auto, Poll interval: 2 seconds
   Queue statistics: Auto, Poll interval: 10 milliseconds
   Depth threshold high: 1228800 bytes, low: 1024 bytes
   Latency threshold high: 900 microseconds, low: 300 microseconds
Interface     Traffic       Queue        Depth-threshold      Latency-threshold
            Statistics    Statistics      High      Low         High      Low
                                             (bytes)           (microseconds)
xe-0/0/1     Enabled       Auto          1228800   1024         900       300
xe-0/0/7     Auto          Auto          1228800   1024         900       300
xe-0/0/8     Auto          Auto          1228800   1024         900       300
```

Verify Streaming Servers Configuration

Purpose

Verify the configuration for streaming data to remote servers is working.

Action

From operational mode, enter the `show analytics streaming-servers` command to display the streaming servers configuration.

```
user@host> show analytics streaming-servers

Address        Port     Stream-Format    Stream-Type   State         Sent
10.94.198.11   50001    json             QS            Established   1100
10.94.198.11   50005    csv              TS/QS         In Progress      0
```

Meaning

The output displays information about the remote streaming server.

Verify Queue Statistics

Purpose

Verify that queue statistics collection is working.

Action

From operational mode, enter the `show analytics queue-statistics` command to display the queue statistics.

```
user@host> show analytics queue-statistics

Time               Interface   Queue-length (bytes)   Latency (us)
Apr 6 0:17:18.224  xe-0/0/1            1043952            835
Apr 6 0:17:18.234  xe-0/0/1            1053520            842
Apr 6 0:17:18.244  xe-0/0/1            1055184            844
```

Meaning

The output displays queue-statistics information as expected.

Verify Traffic Statistics

Purpose

Verify that traffic statistics collection is working.

Action

From operational mode, enter the `show analytics traffic-statistics` command to display the traffic statistics.

```
user@host> show analytics traffic-statistics

Time: Apr 5 19:52:48.549, Physical interface: xe-0/0/1
  Traffic Statistics:                Receive            Transmit
  Total octets:                4797548752936        408886273632
  Total packet:                   5658257464          3190613435
  Octets per second:                       0                   0
  Packet per second:                       0                   0
  Octets dropped:                          0           252901000
  Packet dropped:                          0              252901
  Utilization:                          0.0%                0.0%
Time: Apr 5 19:52:48.549, Physical interface: xe-0/0/7
  Traffic Statistics:                Receive            Transmit
  Total octets:                4790866253100           477139024
```

```
  Total packet:                  5624473639              477944
  Octets per second:                    0                   0
  Packet per second:                    0                   0
  Octets dropped:                       0           166582000
  Packet dropped:                       0              166582
  Utilization:                       0.0%                0.0%
Time: Apr 5 19:52:48.549, Physical interface: xe-0/0/8
  Traffic Statistics:               Receive            Transmit
  Total octets:               4789797668456           764910024
  Total packet:                  5623280870              765715
  Octets per second:                    0                   0
  Packet per second:                    0                   0
  Octets dropped:                       0           156099000
  Packet dropped:                       0              156099
  Utilization:                       0.0%                0.0%
```

Meaning

The output displays traffic-statistics information as expected.

Example: Configure Queue and Traffic Monitoring

IN THIS SECTION

- Requirements | **969**
- Overview | **970**
- Configuration | **971**
- Verification | **978**

This example shows how to configure the enhanced network analytics feature, including queue and traffic monitoring.

Requirements

This example uses the following hardware and software components:

- A QFX5100 standalone switch
- A external streaming server to collect data
- Junos OS Release 13.2X51-D15 software
- TCP server software (for remote streaming servers)

Before you configure network analytics, be sure you have:

- Junos OS Release 13.2X51-D15 or later software installed and running on the QFX5100 switch.
- (Optional for streaming servers for the JSON, CSV, and TSV formats) TCP or UDP server software set up for processing records separated by a newline character (\n) on the remote streaming server.
- (Optional for streaming servers for the GPB format) TCP or UDP build streaming server using the **analytics.proto** file.
- All other network devices running.

Overview

IN THIS SECTION

- Topology | **971**

The network analytics feature provides visibility into the performance and behavior of the data center infrastructure. This feature collects data from the switch, analyzes the data using sophisticated algorithms, and captures the results in reports. Network administrators can use the reports to help troubleshoot problems, make decisions, and adjust resources as needed.

You enable network analytics by first defining a resource profile template, and then applying the profile to the system (for a global configuration) or to individual interfaces.

NOTE: Disabling of the queue or traffic monitoring supersedes the configuration (enabling) of this feature. You disable monitoring by applying a resource profile that includes the `no-queue-monitoring` or `no-traffic-monitoring` configuration statement at the `[edit services analytics resource-profiles]` hierarchy level.

Topology

In this example, the QFX5100 switch is connected to an external server used for streaming statistics data.

Configuration

IN THIS SECTION

- CLI Quick Configuration | **971**
- Configure the Polling Interval for Queue and Traffic Monitoring | **972**
- Configure a Local Statistics File | **972**
- Configure and Apply a Resource Profile for the System | **973**
- Configure and Apply a Resource Profile for an Interface | **974**
- Configure an Export Profile and Collector for Streaming Data | **974**

To configure the network analytics features, perform these tasks:

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them in a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `[edit]` hierarchy level.

```
[edit]
set services analytics resource system polling-interval queue-monitoring 1000
set services analytics resource system polling-interval traffic-monitoring 5
set services analytics collector local file an.stats
set services analytics collector local file an files 3
set services analytics collector local file an size 10m
set services analytics resource-profiles sys-rp queue-monitoring
set services analytics resource-profiles sys-rp traffic-monitoring
set services analytics resource-profiles sys-rp depth-threshold high 999999 low 99
set services analytics resource system resource-profile sys-rp
set services analytics resource-profiles if-rp queue-monitoring
set services analytics resource-profiles if-rp traffic-monitoring
set services analytics resource-profiles if-rp latency-threshold high 2300 low 20
set services analytics resource interfaces xe-0/0/16 resource-profile if-rp
```

```
set services analytics resource interfaces xe-0/0/18 resource-profile if-rp
set services analytics resource interfaces xe-0/0/19 resource-profile if-rp
set services analytics export-profiles ep stream-format gpb
set services analytics export-profiles ep interface information
set services analytics export-profiles ep interface statistics queue
set services analytics export-profiles ep interface statistics traffic
set services analytics export-profiles ep interface status link
set services analytics export-profiles ep system information
set services analytics export-profiles ep system status queue
set services analytics export-profiles ep system status traffic
set services analytics collector address 10.94.198.11 port 50001 transport tcp export-profile ep
set services analytics collector address 10.94.184.25 port 50013 transport udp export-profile ep
```

Configure the Polling Interval for Queue and Traffic Monitoring

Step-by-Step Procedure

To configure the polling interval queue and traffic monitoring globally:

1. Configure the queue monitoring polling interval (in milliseconds) for the system:

```
[edit]
set services analytics resource system polling-interval queue-monitoring 1000
```

2. Configure the traffic monitoring polling interval (in seconds) for the system:

```
[edit]
set services analytics resource system polling-interval traffic-monitoring 5
```

Configure a Local Statistics File

Step-by-Step Procedure

To configure a file for local statistics collection:

1. Configure the filename:

```
[edit]
set services analytics collector local file an.stats
```

2. Configure the number of files:

```
[edit]
set services analytics collector local file an files 3
```

3. Configure the file size:

```
[edit]
set services analytics collector local file an size 10m
```

Configure and Apply a Resource Profile for the System

Step-by-Step Procedure

To define a resource profile template for queue and traffic monitoring resources:

1. Configure a resource profile and enable queue monitoring:

```
[edit]
set services analytics resource-profiles sys-rp queue-monitoring
```

2. Enable traffic monitoring in the profile:

```
[edit]
set services analytics resource-profiles sys-rp traffic-monitoring
```

3. Configure the depth-threshold (high and low values) for queue monitoring in the profile:

```
[edit]
set services analytics resource-profiles sys-rp depth-threshold high 999999 low 99
```

4. Apply the resource profile template to the system resource type for a global configuration:

```
[edit]
set services analytics resource system resource-profile sys-rp
```

Configure and Apply a Resource Profile for an Interface

Step-by-Step Procedure

You can configure queue and traffic monitoring for one or more specific interfaces. The interface-specific configuration supersedes the global (system) configuration. To define a resource profile template for queue and traffic monitoring resources for an interface:

1. Configure a resource profile and enable queue monitoring:

```
[edit]
set services analytics resource-profiles if-rp queue-monitoring
```

2. Enable traffic monitoring in the profile:

```
[edit]
set services analytics resource-profiles if-rp traffic-monitoring
```

3. Configure the latency-threshold (high and low values) for queue monitoring in the profile:

```
[edit]
set services analytics resource-profiles if-rp latency-threshold high 2300 low 20
```

4. Apply the resource profile template to the interfaces resource type for specific interfaces:

```
[edit]
set services analytics resource interfaces xe-0/0/16 resource-profile if-rp
set services analytics resource interfaces xe-0/0/18 resource-profile if-rp
set services analytics resource interfaces xe-0/0/19 resource-profile if-rp
```

Configure an Export Profile and Collector for Streaming Data

Step-by-Step Procedure

To configure a collector (streaming server) for receiving monitoring data:

1. Create an export profile and specify the stream format:

```
[edit]
set services analytics export-profiles ep stream-format gpb
```

2. Configure the export profile to include interface information:

```
[edit]
set services analytics export-profiles ep interface information
```

3. Configure the export profile to include interface queue statistics:

```
[edit]
set services analytics export-profiles ep interface statistics queue
```

4. Configure the export profile to include interface traffic statistics:

```
[edit]
set services analytics export-profiles ep interface statistics traffic
```

5. Configure the export profile to include interface status link information:

```
[edit]
set services analytics export-profiles ep interface status link
```

6. Configure the export profile to include system information:

```
[edit]
set services analytics export-profiles ep system information
```

7. Configure the export profile to include system queue status:

```
[edit]
set services analytics export-profiles ep system status queue
```

8. Configure the export profile to include system traffic status:

```
[edit]
set services analytics export-profiles ep system status traffic
```

9. Configure the transport protocol for the collector addresses and apply an export profile:

```
[edit]
set services analytics collector address 10.94.198.11 port 50001 transport tcp export-profile
ep
set services analytics collector address 10.94.184.25 port 50013 transport udp export-profile
ep
```

NOTE: If you configure the `tcp` or `udp` option for the JSON, CSV, and TSV formats, you must also set up the TCP or UDP client software on the remote collector to process records that are separated by the newline character (\n) on the remote server.

If you configure the `tcp` or `udp` option for the GPB format, you must also set up the TCP or UDP build streaming server using the **analytics.proto** file.

Results

Display the results of the configuration:

```
[edit services analytics]
user@switch# run show configuration
services {
    analytics {
        export-profiles {
            ep {
                stream-format gpb;
                interface {
                    information;
                    statistics {
                        traffic;
                        queue;
                    }
                    status {
                        link;
```

```
                }
            }
            system {
                information;
                status {
                    traffic;
                    queue;
                }
            }
        }
    }
    resource-profiles {
        sys-rp {
            queue-monitoring;
            traffic-monitoring;
            depth-threshold high 99999 low 99;
        }
        if-rp {
            queue-monitoring;
            traffic-monitoring;
            latency-threshold high 2300 low 20;
        }
    }
    resource {
        system {
            resource-profile sys-rp;
            polling-interval {
                traffic-monitoring 5;
                queue-monitoring 1000;
            }
        }
        interfaces {
            xe-0/0/16 {
                resource-profile if-rp;
            }
            xe-0/0/18 {
                resource-profile if-rp;
            }
            xe-0/0/19 {
                resource-profile if-rp;
            }
        }
    }
```

```
        collector {
            local {
                file an size 10m files 3;
            }
            address 10.94.184.25 {
                port 50013 {
                    transport udp {
                        export-profile ep;
                    }
                }
            }
            address 10.94.198.11 {
                port 50001 {
                    transport tcp {
                        export-profile ep;
                    }
                }
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verify the Network Analytics Configuration | **978**
- Verify the Network Analytics Status | **979**
- Verify the Collector Configuration | **980**

Confirm that the configuration is correct and works as expected by performing these tasks:

Verify the Network Analytics Configuration

Purpose

Verify the configuration for network analytics.

Action

From operational mode, enter the `show analytics configuration` command to display the traffic and queue monitoring configuration.

```
user@host> show analytics configuration

Traffic monitoring status is enabled
Traffic monitoring polling interval : 5 seconds
Queue monitoring status is enabled
Queue monitoring polling interval : 1000 milliseconds
Queue depth high threshold : 99999 bytes
Queue depth low threshold : 99 bytes

Interface     Traffic       Queue        Queue depth              Latency
            Statistics    Statistics      threshold              threshold
                                        High         Low      High      Low
                                           (bytes)           (nanoseconds)
xe-0/0/16   enabled       enabled        n/a          n/a      2300      20
xe-0/0/18   enabled       enabled        n/a          n/a      2300      20
xe-0/0/19   enabled       enabled        n/a          n/a      2300      20
```

Meaning

The output displays the traffic and queue monitoring configuration information on the switch.

Verify the Network Analytics Status

Purpose

Verify the network analytics operational status of the switch.

Action

From operational mode, enter the `show analytics status global` command to display global traffic and queue monitoring status.

```
user@host> show analytics status global

Traffic monitoring status is enabled
```

```
Traffic monitoring pollng interval : 5 seconds
Queue monitoring status is enabled
Queue monitoring polling interval : 1000 milliseconds
Queue depth high threshold : 99999 bytes
Queue depth low threshold : 99 bytes
```

From operational mode, enter the `show analytics status` command to display both the interface and global queue monitoring status.

```
user@host> show analytics status

Traffic monitoring status is enabled
Traffic monitoring pollng interval : 5 seconds
Queue monitoring status is enabled
Queue monitoring polling interval : 1000 milliseconds
Queue depth high threshold : 99999 bytes
Queue depth low threshold : 99 bytes

Interface     Traffic    Queue      Queue depth            Latency
            Statistics Statistics    threshold            threshold
                                   High       Low        High      Low
                                      (bytes)            (nanoseconds)
xe-0/0/16   enabled    enabled     n/a        n/a        2300      20
xe-0/0/18   enabled    enabled     n/a        n/a        2300      20
xe-0/0/19   enabled    enabled     n/a        n/a        2300      20
```

Meaning

The output displays the global and interface status of traffic and queue monitoring on the switch.

Verify the Collector Configuration

Purpose

Action

Verify the configuration for the collector for streamed data is working.

From operational mode, enter the `show analytics collector` command to display the streaming servers configuration.

```
user@host> show analytics collector

Address        Port   Transport  Stream format  State          Sent
10.94.184.25   50013  udp        gpb            n/a            484
10.94.198.11   50001  tcp        gpb            In progress    0
```

Meaning

The output displays the collector configuration.

NOTE: The connection state of a port configured with the `udp` transport protocol is always displayed as `n/a`.

Release History Table

Release	Description
13.2X51-D15	In Junos OS Release 13.2X51-D15, the network analytics feature was enhanced, and extensive changes were made to the CLI statements and hierarchies.
13.2X51-D15	Beginning in Junos OS Release 13.2X51-D15, the network analytics feature provides the following enhancements:
13.2X51-D15	Beginning in Junos OS Release 13.2X51-D15, enhancements to the network analytics feature result in changes in the CLI when you configure the feature.
13.2X51-D15	Starting in Junos OS Release 13.2X51-D15, network analytics supports the following streaming data formats and output:
13.2X51-D15	Beginning in Junos OS Release 13.2X51-D15, the traffic and queue monitoring statistics can be stored locally in a single file.
13.2X51-D15	The procedure to configure queue monitoring on a QFX Series standalone switch requires Junos OS Release 13.2X51-D15 or later to be installed on your device.
13.2X51-D15	The procedure to configure a local file for storing queue and traffic monitoring statistics requires Junos OS Release 13.2X51-D15 or later to be installed on your device.

13.2X51-D15	The procedure to configure a collector for receiving streamed analytics data requires Junos OS Release 13.2X51-D15 or later to be installed on your device.

9

PART

Port Mirroring

Port Mirroring and Analyzers | 984

CHAPTER 12

Port Mirroring and Analyzers

IN THIS CHAPTER

- Port Mirroring and Analyzers | **984**
- Configuring Port Mirroring and Analyzers | **1018**
- Configuring Port Mirroring Instances | **1124**
- Configuring Port Mirroring on Physical Interfaces | **1135**
- Configuring Port Mirroring on Logical Interfaces | **1150**
- Configuring Port Mirroring for Multiple Destinations | **1185**
- Configuring Port Mirroring for Remote Destinations | **1197**
- Configuring Port Mirroring Local and Remote Analysis | **1210**
- 1:N Port Mirroring to Multiple Destinations on Switches | **1234**
- Monitoring Port Mirroring | **1238**
- Configure Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic | **1239**
- Troubleshooting Port Mirroring | **1248**

Port Mirroring and Analyzers

SUMMARY

This section describes how port mirroring sends network traffic to analyzer applications.

IN THIS SECTION

- Understanding Port Mirroring and Analyzers | **985**
- Port Mirroring on EX2300, EX3400, and EX4300 Switches | **1000**

- Port Mirroring on ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, EX6200, and EX8200 Series Switches | **1005**
- Port Mirroring on SRX Series Firewalls | **1011**
- Understanding Layer 2 Port Mirroring | **1012**
- Layer 2 Port Mirroring Properties | **1013**
- Application of Layer 2 Port Mirroring Types | **1014**
- Restrictions on Layer 2 Port Mirroring | **1017**

Understanding Port Mirroring and Analyzers

IN THIS SECTION

- Port Mirroring and Analyzer Terms and Definitions | **987**
- Instance Types | **991**
- Port Mirroring and STP | **992**
- Constraints and Limitations | **993**
- Port Mirroring on QFX10000 Series Switches | **997**
- Port Mirroring on QFabric | **998**
- Port Mirroring on OCX Series Switches | **1000**

Port mirroring and analyzers send network traffic to devices running analyzer applications. A port mirror copies Layer 3 IP traffic to an interface. An analyzer copies bridged (Layer 2) packets to an interface. Mirrored traffic can be sourced from single or multiple interfaces. You can use a device attached to a mirror output interface running an analyzer application to perform tasks such as monitoring compliance, enforcing policies, detecting intrusions, monitoring network performance, correlating events, and other problems on the network.

On routers containing an Internet Processor II application-specific integrated circuit (ASIC) or T Series Internet Processor, port mirroring copies Unicast packets entering or exiting a port or entering a VLAN and sends those copies to a local interface for local monitoring or to a VLAN for remote monitoring. The mirrored traffic is received by applications that help you analyze that traffic.

Port mirroring is different from traffic sampling. In traffic sampling, a sampling key based on the IPv4 header is sent to the Routing Engine, where a key is placed in a file or cflowd. Packets based on that key are sent to a cflowd server. In port mirroring, the entire packet is copied and sent out through the specified interface where it can be captured and analyzed in detail.

Use port mirroring to send traffic to devices that analyze traffic for purposes such as monitoring compliance, enforcing policies, detecting intrusions, monitoring and predicting traffic patterns, correlating events, and so on. Port mirroring is needed when you want to perform traffic analysis because a switch normally sends packets only to the port to which the destination device is connected. You probably do not want to send the original packets for analysis before they are forwarded because of the delay that this would cause, so the common alternative is to configure port mirroring to send copies of unicast traffic to another interface and run an analyzer application on a device connected to that interface. .

To configure port mirroring, configure a port-mirroring instance. but don't specify an input for it. Instead, create a firewall filter that specifies the required traffic, and directs it to the instance. Use the `port-mirror` action in a `then` term of the filter for this. The firewall filter must be configured as `family inet`.

Keep performance in mind when configuring port mirroring. Configuring the firewall filter to mirror only the necessary packets reduces the possibility of a performance impact.

You can configure an analyzer statement to define both the input traffic and output traffic in the same analyzer configuration. The traffic to be analyzed can be traffic that enters or exits an interface, or traffic that enters a VLAN. The analyzer configuration enables you to send this traffic to an output interface, instance, or VLAN. You can configure an analyzer at the `[edit forwarding-options analyzer]` hierarchy.

NOTE: On EX Series switches, when you disable any interface in a remote port mirroring VLAN, you will need to re-enable the disabled interface and reconfigure the analyzer session to resume port mirroring.

You can use port mirroring to copy:

- All of the packets entering or exiting an interface in any combination. Copies of packets entering some interfaces and packets exiting other interfaces can be sent to the same local interface or VLAN. If you configure port mirroring to copy packets exiting an interface, traffic that *originates* on that switch or Node device (in a QFabric system) is not copied when it egresses. Only *switched* traffic is copied on egress. (See the limitation on egress mirroring below.)
- Any or all packets entering a VLAN. You cannot use port mirroring to copy packets exiting a VLAN.
- A firewall-filtered sample of packets entering a port or VLAN.
- Firewall filters are not supported on egress ports; that is, you cannot specify policy-based sampling of packets exiting an interface

- In VXLAN environments, firewall-filter based port-mirroring is not supported on core- or spine-facing interfaces.

You can configure both traffic sampling and port mirroring, setting an independent sampling rate and run-length for port-mirrored packets. However, if a packet is selected for both traffic sampling and port mirroring, only port mirroring is executed, as it takes precedence. In other words, if you configure an interface to traffic sample every packet input to the interface and port mirroring also selects that packet to be copied and sent to the destination port, only the port mirroring process is executed. Traffic sampled packets that are not selected for port mirroring continue to be sampled and forwarded to the cflowd server.

Port Mirroring and Analyzer Terms and Definitions

The following tables provide terms and definitions for the port mirroring and analyzer documentation.

Table 116: Terminology

Term	Definition
Analyzer	For EX2300, EX3400, or EX4300 switches, in a mirroring configuration (analyzer) on an the analyzer includes: • The name of the analyzer • Source (input) ports or VLAN (optional)
Analyzer instance	Port-mirroring configuration that includes a name, source interfaces or source VLAN, and a destination for mirrored packets (either a local interface or a VLAN).

Analyzer output interface (also known as monitor port)	Interface to which mirrored traffic is sent and to which a protocol analyzer application is connected. For EX2300, EX3400, and EX4300 Switches, Interfaces used as output for an analyzer must be configured as family ethernet-switching. In addition, the following limitations for analyzer output interfaces apply: • Cannot also be a source port. • Cannot be used for switching. • Do not participate in Layer 2 protocols, such as Spanning Tree Protocol (STP), when part of a port mirroring configuration. • If the bandwidth of the analyzer output interface is not sufficient to handle the traffic from the source ports, overflow packets are dropped.
Analyzer VLAN (also known as monitor VLAN)	VLAN to which mirrored traffic is sent. The mirrored traffic can be used by a protocol analyzer application. The member interfaces in the monitor VLAN are spread across the switches in your network.
Bridge-domain-based analyzer	An analyzer session configured to use bridge domains for input, output or both.
Default analyzer	An analyzer with default mirroring parameters. By default, the mirroring rate is 1 and the maximum packet length is the length of the complete packet.
Global port mirror	A port mirroring configuration that does not have an instance name. The firewall filter action port-mirror will be the action for the firewall filter configuration.
Input interface (also known as mirrored or monitored interface)	An interface that copies traffic to the mirror interface. This traffic can be entering or exiting (ingress or egress) the interface. A mirrored input interface cannot be used as an output interface to the analyzer device.
LAG-based analyzer	An analyzer that has a link aggregation group (LAG) specified as the input (ingress) interface in the analyzer configuration.
Local port mirroring	A port-mirroring configuration where the mirrored packets are copied to an interface on the same switch.
Monitoring station	A computer running a protocol analyzer application.

Next-hop based analyzer	An analyzer configuration that uses the next-hop group as the output to an analyzer.
Native analyzer session	An analyzer session that has both input and output definitions in its analyzer configuration.
Policy-based mirroring	Mirroring of packets that match a firewall filter term. The action `analyzer` *`analyzer-name`* is used in the firewall filter to send specified packets to the analyzer.
Port-based analyzer	An analyzer session whose configuration defines interfaces for both input and output.
Port mirroring instance	A port-mirroring configuration that does not specify an input source; it specifies only an output destination. A firewall filter configuration must be defined for the input source. A firewall filter configuration must be defined to mirror packets that match the match conditions defined in the firewall filter term. The action item port-mirror-instance instance-name in the firewall filter configuration is used to send packets to the analyzer and these packets form the input source. Use the `port-mirror-instance` *`instance-name`* action in the firewall filter configuration to send packets to the port mirror. **NOTE**: Port mirroring instance is not supported on NFX150 devices.
Protocol analyzer application	An application used to examine packets transmitted across a network segment. Also commonly called network analyzer, packet sniffer, or probe.

Output interface (also known as the monitor interface)	The interface to where the copies of packets are sent and to which a device running an analyzer is connected. The following limitations apply to an output interface (the target mirror interface): • Cannot also be a source port. • Cannot be used for switching. • Cannot be an aggregated Ethernet interface (LAG). • Cannot participate in Layer 2 protocols, such as Spanning Tree Protocol (STP). • Existing VLAN associations are lost when port mirroring is applied to the interface. • Packets are dropped if the capacity of the output interface is insufficient to handle the traffic from the mirrored source ports.
Output IP address	IP address of the device running an analyzer application. The device can be on a remote network. When you use this feature: • Mirrored packets are GRE-encapsulated. The analyzer application must be able to de-encapsulate GRE-encapsulated packets or the GRE-encapsulated packets must be de-encapsulated before reaching the analyzer application. (You can use a network sniffer to de-encapsulate the packets.) • The output IP address cannot be in the same subnetwork as any of the switch management interfaces. • If you create virtual routing instances and an analyzer configuration that includes an output IP address, the output IP address belongs to the default virtual routing instance (inet.0 routing table).

Output VLAN (also known as monitor or analyzer VLAN)	VLAN to where copies of the packets are sent and to where a device running an analyzer is connected. The analyzer VLAN can span multiple switches. The following limitations apply to an output VLAN: • Cannot be a private VLAN or VLAN range. • Cannot be shared by multiple `analyzer` statements. • Cannot be a member of any other VLAN. • Cannot be an aggregated Ethernet interface (LAG). • On some switches, only one interface can be a member of the analyzer VLAN. This limitation does not apply on the QFX10000 switch. When *ingress* traffic is mirrored, multiple QFX10000 interfaces can belong to the output VLAN and traffic is mirrored from all of those interfaces. If *egress* traffic is mirrored on a QFX10000 switch, only one interface can be a member of the analyzer VLAN.
Remote port mirroring	Functions the same as local port mirroring, except that the mirrored traffic is not copied to a local analyzer port but is flooded to an analyzer VLAN that you create specifically for the purpose of receiving mirrored traffic. You cannot send mirrored packets to a remote IP address on a QFabric system.
VLAN-based analyzer	An analyzer session whose configuration uses VLANs for both input and output or for either input or output.

SEE ALSO

Port Mirroring and Analyzers | 984

Instance Types

To configure port mirroring, configure an instance of one of the following types:

- Analyzer instance—Specify the input and output for the instance. This instance type is useful for ensuring that all traffic transiting an interface or entering a VLAN is mirrored and sent to the analyzer.

- Port-mirroring instance—You create a firewall filter that identifies the desired traffic and copies it to the mirror port. You do not specify an input for this instance type. This instance type is useful for controlling the types of traffic that are mirrored. You can direct traffic to it in the following ways:
  - Specify the name of the port-mirroring instance in the firewall filter by using the `port-mirror-instance` *`instance-name`* action when there are multiple port-mirroring instances defined.
  - Send the mirrored packets to the output interface defined in the instance by using the `port-mirror` action when there is only one port-mirroring instance defined.

For QFX5100, QFX5110, QFX5120, QFX5200, QFX5210, EX4600 and EX4650 switches, the following port mirroring guidelines apply:

- A maximum of four port mirroring instances, or four analyzer sessions, can be configured at the same time. In other words, you cannot configure four port mirroring instances *and* four analyzer sessions together.
- If there are no port mirroring instances, (that is, only analyzer sessions are configured), then you can enable up to three analyzer sessions for ingress and egress mirroring. The remaining analyzer session must be used for ingress mirroring only.
- If you have only one port mirroring instance configured, then of the remaining instances, you can configure up to three analyzers for ingress mirroring, and two analyzers for egress mirroring.
- If you have two port mirroring instance configured, then of the remaining instances, you can configure up to two analyzers for ingress mirroring, and one analyzer for egress mirroring.
- If you have three port mirroring instance configured, then the remaining instance can only be configured as an analyzer (for either ingress or egress mirroring),

Port Mirroring and STP

The behavior of STP in a port-mirroring configuration depends on the version of Junos OS you are using:

- Junos OS 13.2X50, Junos OS 13.2X51-D25 or earlier, Junos OS 13.2X52: When STP is enabled, port mirroring might not succeed because STP might block the mirrored packets.
- Junos OS 13.2X51-D30, Junos OS 14.1X53: STP is disabled for mirrored traffic. You must ensure that your topology prevents loops of this traffic.

Constraints and Limitations

IN THIS SECTION

- Constraints and Limitations for QFX5100 and QFX5200 Switches | **996**

The following constraints and limitations apply to port mirroring:

Mirroring only the packets required for analysis reduces the possibility of reducing overall performance. If you mirror traffic from multiple ports, the mirrored traffic might exceed the capacity of the output interface. The overflow packets are dropped. We recommend that you limit the amount of mirrored traffic by selecting specific interfaces and avoid using the `all` keyword. You can also limit the amount of mirrored traffic by using a *firewall filter* to send specific traffic to the port mirroring instance.

- You can create a total of four port-mirroring configurations.
- On EX9200 switches, port mirroring is **not** supported on EX9200-15C line cards.
- Each Node group in a QFabric system is subject to the following constraints:
  - Up to four of the configurations can be used for local port mirroring.
  - Up to three of the configurations can be used for remote port mirroring.
- Regardless of whether you are configuring a standalone switch or a Node group:
  - There can be no more than two configurations that mirror ingress traffic. If you configure a firewall filter to send mirrored traffic to a port, this counts as an ingress mirroring configuration for the switch or Node group to which the filter is applied.
  - There can be no more than two configurations that mirror egress traffic.
  - On QFabric systems, there is no system-wide limit on the total number of mirror sessions.
- You can configure only one type of output in one port-mirroring configuration to complete a `set analyzer` *`name`* `output` statement:
  - `interface`
  - `ip-address`
  - `vlan`
- Configure mirroring in an analyzer (with `set forwarding-options analyzer`) on only one logical interface for the same physical interface. If you try to configure mirroring on multiple logical interfaces

configured on a physical interface, only the first logical interface is successfully configured; the remaining logical interfaces return configuration errors.

- If you mirror egress packets, do not configure more than 2000 VLANs on a standalone switch or QFabric system. If you do, some VLAN packets might contain incorrect VLAN IDs. This applies to any VLAN packets, not just the mirrored copies.
- The `ratio` and `loss-priority` options are not supported.
- Packets with physical layer errors are not sent to the output port or VLAN.
- If you use sFlow monitoring to sample traffic, it does not sample the mirror copies when they exit the output interface.
- You cannot mirror packets exiting or entering the following ports:
  - Dedicated Virtual Chassis interfaces
  - Management interfaces (me0 or vme0)
  - Fibre Channel interfaces
  - Integrated routing and bridging (IRB) interfaces (also known as routed VLAN interfaces or RVIs)
- In a port-mirroring instance, you cannot configure an inet or inet6 interface as the output interface. The following switches do not support the `set forwarding-options port-mirroring instance <instance-name> family inet output interface <interface-name>` configuration:

  Table 117: Switches Not Supporting family inet/inet6 as Output Interface

EX Switches	QFX Switches
EX2300	QFX3500
EX3400	QFX5100
EX4100	QFX5110
EX4300	QFX5120
EX4400	QFX5130

Table 117: Switches Not Supporting family inet/inet6 as Output Interface *(Continued)*

EX Switches	QFX Switches
EX4600	QFX5200
EX4650	QFX5210
	QFX5220
	QFX5700

- An aggregated Ethernet interface cannot be an output interface if the input is a VLAN or if traffic is sent to the analyzer by using a firewall filter.
- When mirrored packets are sent out of an output interface, they are not modified for any changes that might be applied to the original packets on egress, such as CoS rewriting.
- An interface can be the input interface for only one mirroring configuration. Do not use the same interface as the input interface for multiple mirroring configurations.
- CPU-generated packets (such as ARP, ICMP, BPDU, and LACP packets) cannot be mirrored on egress.
- VLAN-based mirroring is not supported for STP traffic.
- (QFabric systems only) If you configure a QFabric analyzer to mirror egress traffic and the input and output interfaces are on different Node devices, the mirrored copies will have incorrect VLAN IDs.

  This limitation does not apply if you configure a QFabric analyzer to mirror egress traffic and the input and output interfaces are on the same Node device. In this case the mirrored copies will have the correct VLAN IDs (as long as you do not configure more than 2000 VLANs on the QFabric system).
- True egress mirroring is defined as mirroring the exact number of copies and the exact packet modifications that went out the egress port. Because the processors on QFX5100 and EX4600 switches implement egress mirroring in the ingress pipeline, those switches do not provide accurate egress packet modifications, so egress mirrored traffic can carry incorrect VLAN tags that differ from the tags in the original traffic.
- If you configure a port-mirroring instance to mirror traffic exiting an interface that performs VLAN encapsulation, the source and destination MAC addresses of the mirrored packets are not the same as those of the original packets.

- Mirroring on member interfaces of a LAG is not supported.
- Egress VLAN mirroring is not supported.

The following constraints and limitations apply to remote port mirroring:

- If you configure an output IP address, that address cannot be in the same subnetwork as any of the switch management interfaces.
- If you create virtual routing instances and you create an analyzer configuration that includes an output IP address, the output IP address belongs to the default virtual routing instance (inet.0 routing table).
- An output VLAN cannot be a private VLAN or VLAN range.
- An output VLAN cannot be shared by multiple analyzer sessions or port-mirror instances.
- An output VLAN interface cannot be a member of any other VLAN.
- An output VLAN interface cannot be an aggregated Ethernet interface.
- If the output VLAN has more than one member interface, then traffic is mirrored only to the first member of the VLAN, and other members of the same VLAN do not carry any mirrored traffic.
- For remote port mirroring to an IP address (GRE encapsulation), if you configure more than one analyzer session or port-mirror instance, and the IP addresses of the analyzers or port-mirror instance are reachable through the same interface, then only one analyzer session or port-mirror instance will be configured.
- The number of possible output interfaces in remote port mirroring varies among the switches in the QFX5K line:
  - QFX5110, QFX5120, QFX5210—Support a maximum of 4 output interfaces
  - QFX5100 and QFX5200—Support a maximum of 3 output interfaces.
- Whenever any member in a remote port mirroring VLAN is removed from that VLAN, reconfigure the analyzer session for that VLAN.

Constraints and Limitations for QFX5100 and QFX5200 Switches

The following considerations apply to port mirroring on QFX5100 and QFX5200 switches:

- When configuring mirroring with output to IP address, the destination IP address should be reachable, and ARP must be resolved.
- ECMP (Equal Cost Multiple Path) load balancing is not supported for mirrored destinations.

- The number of output interfaces in remote port mirroring (RSPAN) varies. For QFX5110, QFX5120, and QFX5210, switches the maximum is four output interfaces. For QFX5100 and QFX5200 switches, the maximum is three.
- When specifying a link aggregation group (LAG) as the mirroring output interface, a maximum of eight interfaces are mirrored.
- The mirroring input can be a LAG, a physical interface with any unit (such as ae0.101 or xe-0/0/0.100), or a sub-interface. In any case, all the traffic on the LAG or physical interface is mirrored.
- You cannot set up an independent mirroring instance on a member interface of a LAG.
- An output interface that is included in one mirroring instance cannot also be used in another mirroring instance.
- In a port-mirroring instance, dropped packets in the egress pipeline of forwarding-path are never-the-less mirrored to the destination. This is because the mirroring action occurs at the ingress pipeline, before the drop action.
- In a port-mirroring instance, only one mirror output destination can be specified.
- Output mirror destinations that are configured across multiple port-mirroring or analyzer instances must all be unique.
- For ERSPAN IPv6 addresses, egress mirroring is not supported when the output to the analyzer/port-mirroring is a remote IPv6 address. Egress mirror is not supported.
- For local mirroring, the output interface must be family ethernet-switching, with or without VLAN (that is, not a Layer 3 interface).
- When configuring a port-mirroring or analyzer instance in a service provider environment, use the VLAN name rather than the VLAN ID.

Port Mirroring on QFX10000 Series Switches

The following list describes constraints and limitations that apply specifically to QFX10000 Series switches. For general information about port mirroring on switches, see earlier sections in this *Port Mirroring and Analyzers* document that do not specifically call out other platform names in the section title.

- Only ingress global port mirroring is supported. You can configure global port mirroring with input parameters such as `rate` , `run-length`, and `maximum-packet-length`. Egress global port mirroring is not supported.
- Port mirroring instances are supported only for remote port mirroring. Port mirroring **global** instances are supported for local mirroring.

- Local port mirroring is supported on these firewall filter families only: `inet` and `inet6`.
- Local port mirroring is not supported on firewall filter families `any` or `ccc`.

Port Mirroring on QFabric

The following constraints and limitations apply to local and remote port mirroring:

- You can create a total of four port-mirroring configurations.
- Each Node group in a QFabric system is subject to the following constraints:
  - Up to four of the configurations can be used for local port mirroring.
  - Up to three of the configurations can be used for remote port mirroring.
- Regardless of whether you are configuring a standalone switch or a Node group:
  - There can be no more than two configurations that mirror ingress traffic. If you configure a firewall filter to send mirrored traffic to a port—that is, you use the `analyzer` action modifier in a filter term—this counts as an ingress mirroring configuration for the switch or Node group to which the filter is applied.
  - There can be no more than two configurations that mirror egress traffic.
  - On QFabric systems, there is no system-wide limit on the total number of mirror sessions.
- You can configure only one type of output in one port-mirroring configuration to complete a `set analyzer` *`name`* `output` statement:
  - `interface`
  - `ip-address`
  - `vlan`
- Configure mirroring in an analyzer (with `set forwarding-options analyzer`) on only one logical interface for the same physical interface. If you try to configure mirroring on multiple logical interfaces configured on a physical interface, only the first logical interface is successfully configured; the remaining logical interfaces return configuration errors.
- If you mirror egress packets, do not configure more than 2000 VLANs on a QFX Series switch. If you do, some VLAN packets might contain incorrect VLAN IDs. This applies to any VLAN packets, not just the mirrored copies.
- The `ratio` and `loss-priority` options are not supported.
- Packets with physical layer errors are not sent to the output port or VLAN.

- If you use sFlow monitoring to sample traffic, it does not sample the mirror copies when they exit the output interface.
- You cannot mirror packets exiting or entering the following ports:
  - Dedicated Virtual Chassis interfaces
  - Management interfaces (me0 or vme0)
  - Fibre Channel interfaces
  - Integrated routing and bridging (IRB) interfaces (also known as routed VLAN interfaces or RVIs)
- An aggregated Ethernet interface cannot be an output interface if the input is a VLAN or if traffic is sent to the analyzer by using a firewall filter.
- When mirrored packets are sent out of an output interface, they are not modified for any changes that might be applied to the original packets on egress, such as CoS rewriting.
- An interface can be the input interface for only one mirroring configuration. Do not use the same interface as the input interface for multiple mirroring configurations.
- CPU-generated packets (such as ARP, ICMP, BPDU, and LACP packets) cannot be mirrored on egress.
- VLAN-based mirroring is not supported for STP traffic.
- (QFabric systems only) If you configure a QFabric analyzer to mirror egress traffic and the input and output interfaces are on different Node devices, the mirrored copies will have incorrect VLAN IDs.

  This limitation does not apply if you configure a QFabric analyzer to mirror egress traffic and the input and output interfaces are on the same Node device. In this case the mirrored copies will have the correct VLAN IDs (as long as you do not configure more than 2000 VLANs on the QFabric system).
- True egress mirroring is defined as mirroring the exact number of copies and the exact packet modifications that went out the egress port. Because the processors on QFX5xxx (including QFX5100, QFX5110, QFX5120, QFX5200, and QFX5210) and EX4600 (including EX4600 and EX4650) switches implement egress mirroring in the ingress pipeline, those switches do not provide accurate egress packet modifications, so egress mirrored traffic can carry incorrect VLAN tags that differ from the tags in the original traffic.
- If you configure a port-mirroring instance to mirror traffic exiting an interface that performs VLAN encapsulation, the source and destination MAC addresses of the mirrored packets are not the same as those of the original packets.
- Mirroring on member interfaces of a LAG is not supported.
- Egress VLAN mirroring is not supported.

Port Mirroring on OCX Series Switches

The following constraints and limitations apply to port mirroring on OCX Series switches:

- You can create a total of four port-mirroring configurations. There can be no more than two configurations that mirror ingress or egress traffic.
- If you use sFlow monitoring to sample traffic, it does not sample the mirror copies when they exit the output interface.
- You can create only one port-mirroring session.
- You cannot mirror packets exiting or entering the following ports:
  - Dedicated Virtual Chassis interfaces
  - Management interfaces (me0 or vme0)
  - Fibre Channel interfaces
  - Routed VLAN interfaces or IRB interfaces
- An aggregated Ethernet interface cannot be an output interface.
- Do not include an 802.1Q subinterface that has a unit number other than 0 in a port mirroring configuration. Port mirroring does not work with subinterfaces if their unit number is not 0. (You configure 802.1Q subinterfaces by using the `vlan-tagging` statement.)
- When packet copies are sent out the output interface, they are not modified for any changes that are normally applied on egress, such as CoS rewriting.
- An interface can be the input interface for only one mirroring configuration. Do not use the same interface as the input interface for multiple mirroring configurations.
- CPU-generated packets (such as ARP, ICMP, BPDU, and LACP packets) cannot be mirrored on egress.
- VLAN-based mirroring is not supported for STP traffic.

Port Mirroring on EX2300, EX3400, and EX4300 Switches

IN THIS SECTION

- Overview | **1001**
- Configuration Guidelines for Port Mirroring and Analyzers on EX2300, EX3400, and EX4300 Switches | **1002**

Mirroring might be needed for traffic analysis on a switch because a switch, unlike a hub, does not broadcast packets to every port on the destination device. The switch sends packets only to the port to which the destination device is connected.

Overview

Junos OS running on EX2300, EX3400, and EX4300 Series switches supports the Enhanced Layer 2 Software (ELS) configurations that facilitate analyzing traffic on these switches at the packet level.

You use port mirroring to copy packets to a local interface for local monitoring or to a VLAN for remote monitoring. You can use analyzers to enforce policies concerning network usage and file sharing, and to identify sources of problems on your network by locating abnormal or heavy bandwidth usage by specific stations or applications.

Port mirroring is configured at the `[edit forwarding-options port-mirroring]` hierarchy level. To mirror routed (Layer 3) packets, you can use the port mirroring configuration in which the `family` statement is set to `inet` or `inet6`.

You can use port mirroring to copy these packets:

- **Packets entering or exiting a port**—You can mirror the packets in any combination of packets entering or exiting ports up to 256 ports.

  In other words, you can send copies of the packets entering some ports and the packets exiting other ports to the same local analyzer port or analyzer VLAN.

- **Packets entering a VLAN**—You can mirror the packets entering a VLAN to either a local analyzer port or to an analyzer VLAN. You can configure up to 256 VLANs, including a VLAN range and PVLANs, as ingress input to an analyzer.

- **Policy-based sample packets**—You can mirror a policy-based sample of packets that are entering a port or a VLAN. You configure a *firewall filter* to establish a policy to select the packets to be mirrored and send the sample to a port-mirroring instance or to an analyzer VLAN.

You can configure port mirroring on the switch to send copies of Unicast traffic to an output destination such as an interface, a routing-instance, or a VLAN. Then, you can analyze the mirrored traffic by using a protocol analyzer application. The protocol analyzer application can run either on a computer connected to the analyzer output interface or on a remote monitoring station. For the input traffic, you can configure a firewall filter term to specify whether port mirroring must be applied to all packets at the interface to which the firewall filter is applied. You can apply a firewall filter configured with the action `port-mirror` or `port-mirror-instance` *`name`* to the input or output logical interfaces (including aggregated Ethernet logical interfaces), to traffic forwarded or flooded to a VLAN, or traffic forwarded or flooded to a VPLS routing instance. EX2300, EX3400, and EX4300 switches support port mirroring of VPLS (`family ethernet-switching` or `family vpls`) traffic and VPN traffic with `family ccc` in a Layer 2 environment.

Within a firewall filter term, you can specify the port-mirroring properties under the `then` statement in the following ways:

- Implicitly reference the port-mirroring properties in effect on the port.
- Explicitly reference a particular named instance of port mirroring.

Configuration Guidelines for Port Mirroring and Analyzers on EX2300, EX3400, and EX4300 Switches

When you configure port mirroring we recommend that you follow certain guidelines to ensure that you obtain optimum benefit from mirroring. Additionally, we recommend that you disable mirroring when you are not using it and that you select specific interfaces for which packets must be mirrored (that is, select specific interfaces as input to the analyzer) in preference to using the `all` keyword option that enables mirroring on all interfaces and can impact overall performance. Mirroring only the necessary packets reduces any potential performance impact.

With local mirroring, traffic from multiple ports is replicated to the analyzer output interface. If the output interface for an analyzer reaches capacity, packets are dropped. Thus, while configuring an analyzer, you must consider whether the traffic being mirrored exceeds the capacity of the analyzer output interface.

You can configure an analyzer at the `[edit forwarding-options analyzer]` hierarchy.

NOTE: True egress mirroring is defined as mirroring the exact number of copies and the exact packet modifications that went out the egress switched port. Because the processor on EX2300 and EX3400 switches implements egress mirroring in the ingress pipeline, those switches do not provide accurate egress packet modifications, so egress mirrored traffic can carry VLAN tags that differ from the tags in the original traffic.

Table 118 on page 1002 summarizes additional configuration guidelines for mirroring on EX2300, EX3400, and EX4300 switches.

Table 118: Configuration Guidelines for Port Mirroring and Analyzers on EX2300, EX3400, and EX4300 Switches

Guideline	Value or Support Information	Comment
Number of VLANs that you can use as ingress input to an analyzer.	256	

Table 118: Configuration Guidelines for Port Mirroring and Analyzers on EX2300, EX3400, and EX4300 Switches *(Continued)*

Guideline	Value or Support Information	Comment
Number of port-mirroring sessions and analyzers that you can enable concurrently.	4	You can configure a total of four sessions and you can enable only one of the following at any point in time: • A maximum of four port-mirroring sessions (including the global port-mirroring session). • A maximum of four analyzer sessions. • A combination of port-mirroring and analyzer sessions, and the total of this combination must be four. You can configure more than the specified number of port-mirroring instances or analyzers on the switch, but you can enable only the specified number for a session.
Types of ports on which you cannot mirror traffic.	• *Virtual Chassis* ports (VCPs) • Management Ethernet ports (me0 or vme0) • Integrated routing and bridging (IRB) interfaces; also known as routed VLAN interfaces (RVIs). • VLAN-tagged Layer 3 interfaces	

Table 118: Configuration Guidelines for Port Mirroring and Analyzers on EX2300, EX3400, and EX4300 Switches *(Continued)*

Guideline	Value or Support Information	Comment
Protocol families that you can include in a port-mirroring configuration for remote traffic.	any	
Traffic directions that you can configure for mirroring on ports in firewall-filter–based configurations.	Ingress and egress	
Mirrored packets exiting an interface that reflect rewritten class-of-service (CoS) DSCP or 802.1p bits.	Applicable	
Packets with physical layer errors.	Applicable	Packets with these errors are filtered out and thus are not sent to the analyzer.
Port mirroring does not support line-rate traffic.	Applicable	Port mirroring for line-rate traffic is done on a best-effort basis.
Mirroring of packets egressing a VLAN.	Not supported	
Port-mirroring or analyzer output on a LAG interface.	Supported	
Maximum number of child members on a port-mirroring or analyzer output LAG interface.	8	
Maximum number of interfaces in a remote port-mirroring or analyzer VLAN.	1	
Egress mirroring of host-generated control packets.	Not Supported	

Table 118: Configuration Guidelines for Port Mirroring and Analyzers on EX2300, EX3400, and EX4300 Switches *(Continued)*

Guideline	Value or Support Information	Comment
Configuring Layer 3 logical interfaces in the `input` stanza of an analyzer.	Not supported	This functionality can be achieved by configuring port mirroring.
The analyzer input and output stanzas containing members of the same VLAN or the VLAN itself must be avoided.	Applicable	

Port Mirroring on ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, EX6200, and EX8200 Series Switches

IN THIS SECTION

- Overview | **1006**
- Configuration Guidelines for ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, EX6200, and EX8200 Series Switches | **1007**

Juniper Networks Junos operating system (Junos OS) running on ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, EX6200 or EX8200 Series switches does not support Enhanced Layer 2 Software (ELS) configurations. As such, Junos OS does not include the `port-mirroring` statement found at the `edit forwarding-options` level of the hierarchy of other Junos OS packages, or the `port-mirror` action in firewall filter terms.

You can use *port mirroring* to facilitate analyzing traffic on your Juniper Networks EX Series Ethernet Switch on a packet level. You might use port mirroring as part of monitoring switch traffic for such purposes as enforcing policies concerning network usage and file sharing and for identifying sources of problems on your network by locating abnormal or heavy bandwidth usage by particular stations or applications.

You can use port mirroring to copy these packets to a local interface or to a VLAN:

- Packets entering or exiting a port

- You can send copies of the packets entering some ports and the packets exiting other ports to the same local analyzer port or analyzer VLAN.
- Packets entering a VLAN on ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, or EX6200 switches
- Packets exiting a VLAN on EX8200 switches

Overview

Port mirroring is used for traffic analysis on a switch because a switch, unlike a hub, does not broadcast packets to every port on the destination device. The switch sends packets only to the port to which the destination device is connected.

You configure port mirroring on the switch to send copies of Unicast traffic to either a local analyzer port or an analyzer VLAN. Then you can analyze the mirrored traffic by using a protocol analyzer. The protocol analyzer can run either on a computer connected to the analyzer output interface or on a remote monitoring station.

You can use port mirroring to mirror any of the following:

- **Packets entering or exiting a port**—You can mirror the packets in any combination of packets entering or exiting ports up to 256 ports.

  In other words, you can send copies of the packets entering some ports and the packets exiting other ports to the same local analyzer port or analyzer VLAN.

- **Packets entering a VLAN on an ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, or EX6200 switch**—You can mirror the packets entering a VLAN on an analyzer VLAN. On EX3200, EX4200, EX4500, and EX4550 switches, you can configure multiple VLANs (up to 256 VLANs), including a VLAN range and PVLANs, as ingress input to an analyzer.

- **Packets exiting a VLAN on an EX8200 switch**—You can mirror the packets exiting a VLAN on an EX8200 switch to either a local analyzer port or to an analyzer VLAN. You can configure multiple VLANs (up to 256 VLANs), including a VLAN range and PVLANs, as egress input to an analyzer.

- **Statistical samples**—You can mirror a statistical sample of packets that are:

  - Entering or exiting a port
  - Entering a VLAN on an ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, or EX6200 switch
  - Exiting a VLAN on an EX8200 switch

  You specify the sample number of packets by setting the ratio. You can send the sample to either a local analyzer port or to an analyzer VLAN.

- **Policy-based sample**—You can mirror a policy-based sample of packets that are entering a port or a VLAN. You configure a *firewall filter* to establish a policy to select the packets to be mirrored. You can send the sample to a local analyzer port or to an analyzer VLAN.

Configuration Guidelines for ACX7024, ACX7100, ACX7509, EX2200, EX3200, EX3300, EX4200, EX4500, EX4550, EX6200, and EX8200 Series Switches

When you configure port mirroring, we recommend that you follow certain guidelines to ensure that you obtain optimum benefit from the port mirroring feature. Additionally, we recommend that you disable port mirroring when you are not using it and that you select specific interfaces for which packets must be mirrored (that is, select specific interfaces as input to the analyzer) as opposed to using the `all` keyword that enables port mirroring on all interfaces and can impact overall performance. You can also limit the amount of mirrored traffic by using statistical sampling, setting a ratio to select a statistical sample, or using a firewall filter. Mirroring only the necessary packets reduces any potential performance impact.

With local port mirroring, traffic from multiple ports is replicated to the analyzer output interface. If the output interface for an analyzer reaches capacity, packets are dropped. Thus, while configuring an analyzer, you must consider whether the traffic being mirrored exceeds the capacity of the analyzer output interface.

NOTE: On ACX5448 routers, under the [`edit forwarding-options analyzer an input egress`] hierarchy level, analyser input must be configured only on .0 logical interfaces for ingress and egress interfaces. If you configure logical interfaces other than .0, then an error is shown during commit. The following is a sample commit error shown when the analyzer input is configured .100 logical interface:

```
[edit forwarding-options analyzer an input egress]
  'interface ge-0/0/12.100'
    Analyzer input can only be on .0 interfaces
error: configuration check-out failed
```

NOTE:
Feature Explorer

Table 119: Configuration Guidelines

Guideline	Description	Comment
Number of VLANs that you can use as ingress input to an analyzer	• 16 Ingress or 8 Ingress and 8 Egress—ACX7024 devices 1—EX2200 switches • 256—EX3200, EX4200, EX4500, EX4550, and EX6200 switches • Does not apply—EX8200 switches	
Number of analyzers that you can enable concurrently (applies to both standalone switches and to Virtual Chassis)	• 1—EX2200, EX3200, EX4200, EX3300, and EX6200 switches • 7 port-based or 1 global—EX4500 and EX4550 switches • 7 total, with one based on a VLAN, firewall filter, or LAG and with the remaining 6 based on firewall filters—EX8200 switches **NOTE**: An analyzer configured using a firewall filter does not support mirroring of packets that are egressing ports.	• You can *configure* more than the specified number of analyzers on the switch, but you can *enable* only the specified number for a session. Use **disable ethernet-switching-options analyzer name** to disable an analyzer. • See the next row entry in this table for the exception to the number of firewall-filter–based analyzers allowed on EX4500 and EX4550 switches. • On an EX4550 Virtual Chassis, you can configure only one analyzer if ports in the input and output definitions are on different switches in a Virtual Chassis. To configure multiple analyzers, an entire analyzer session must be configured on the same switch of a Virtual Chassis.
Number of firewall-filter–based analyzers that you can configure on EX4500 and EX4550 switches	• 1—EX4500 and EX4550 switches	If you configure multiple analyzers, you cannot attach any of them to a firewall filter.

Table 119: Configuration Guidelines *(Continued)*

Guideline	Description	Comment
Types of ports on which you cannot mirror traffic	• *Virtual Chassis* ports (VCPs) • Management Ethernet ports (me0 or vme0) • Routed VLAN interfaces (RVIs) • VLAN-tagged Layer 3 interfaces	
If port mirroring is configured to mirror packets exiting 10-Gigabit Ethernet ports on EX8200 switches, packets are dropped in both network and mirrored traffic when the mirrored packets exceed 60 percent of the 10-Gigabit Ethernet port traffic.	• EX8200 switches	
Traffic directions for which you can specify a ratio	• Ingress only—EX8200 switches • Ingress and egress—All other switches	
Protocol families that you can include in a firewall-filter-based remote analyzer	• Any except `inet` and `inet6`—EX8200 switches • Any—All other switches	You can use `inet` and `inet6` on EX8200 switches in a local analyzer.
Traffic directions that you can configure for mirroring on ports in firewall-filter–based configurations	• Ingress only—All switches	

Table 119: Configuration Guidelines *(Continued)*

Guideline	Description	Comment
Mirrored packets on tagged interfaces might contain an incorrect VLAN ID or Ethertype.	• Both VLAN ID and Ethertype—EX2200 switches • VLAN ID only—EX3200 and EX4200 switches • Ethertype only—EX4500 and EX4550 switches • Does not apply—EX8200 switches	
Mirrored packets exiting an interface do not reflect rewritten class-of-service (CoS) DSCP or 802.1p bits.	• All switches	
The analyzer appends an incorrect 802.1Q (`dot1q`) header to the mirrored packets on the routed traffic or does not mirror any packets on the routed traffic when an egress VLAN that belongs to a routed VLAN interface (RVI) is configured as the input for that analyzer.	• EX8200 switches • Does not apply—All other switches	As a workaround, configure an analyzer that uses each port (member interface) of the VLAN as egress input.
Packets with physical layer errors are not sent to the local or remote analyzer.	• All switches	Packets with these errors are filtered out and thus are not sent to the analyzer.
Port mirroring configuration on a Layer 3 interface with the output configured to a VLAN is not available on EX8200 switches.	• EX8200 switches • Does not apply—All other switches	

Table 119: Configuration Guidelines *(Continued)*

Guideline	Description	Comment
Port mirroring does not support line-rate traffic.	• All switches	Port mirroring for line-rate traffic is done on a best-effort basis.
In an EX8200 Virtual Chassis, to mirror traffic across the Virtual Chassis, the output port must be a LAG.	• EX8200 Virtual Chassis • Does not apply—All other switches	In an EX8200 Virtual Chassis: • You can configure LAG as a monitor port only for native analyzers. • You cannot configure LAG as a monitor port for analyzers based on firewall filters. • If an analyzer configuration contains LAG as a monitor port, then you cannot configure VLAN in the input definition of an analyzer.
In standalone EX8200 switches, you can configure LAG in the output definition.	• EX8200 standalone switches • Does not apply—All other switches	In EX8200 standalone switches: • You can configure a LAG as a monitor port on both native and firewall-based analyzers. • If a configuration contains LAG as a monitor port, then you cannot configure VLAN in the input definition of an analyzer.

Port Mirroring on SRX Series Firewalls

Port mirroring copies packets entering or exiting a port and sends the copies to a local interface for monitoring. Port mirroring is used to send traffic to applications that analyze traffic for purposes such as monitoring compliance, enforcing policies, detecting intrusions, monitoring and predicting traffic patterns, correlating events, and so on. </para><para>Port mirroring is used to send a copy of all the packets or only the sampled packets seen on a port to a network monitoring connection. You can mirror the packets either on the incoming port (ingress port mirroring) or the outgoing port (egress port mirroring).

Port mirroring is supported only on the SRX Series Firewalls with the following I/O cards:

- SRX1K-SYSIO-GE
- SRX1K-SYSIO-XGE
- SRX3K-SFB-12GE
- SRX3K-2XGE-XFP
- SRX5K-FPC-IOC Flex I/O

On SRX Series Firewalls, all packets passing through the `mirrored` port are copied and sent to the specified `mirror-to` port. These ports must be on the same Broadcom chipset in the I/O cards.

On SRX Series Firewalls, port mirroring works on physical interfaces only.

Understanding Layer 2 Port Mirroring

On routing platforms and switches that contain an Internet Processor II ASIC, you can send a copy of any incoming packet from the routing platform or switch to an external host address or a packet analyzer for analysis. This is known as *port mirroring*.

In Junos OS Release 9.3 and later, Juniper Networks MX Series 5G Universal Routing Platforms in a Layer 2 environment support *port mirroring* for Layer 2 bridging traffic and virtual private LAN service (VPLS) traffic.

In Junos OS Release 9.4 and later, MX Series routers in a Layer 2 environment support port mirroring for Layer 2 VPN traffic over a circuit cross-connect (CCC) that transparently connects logical interfaces of the same type.

In Junos OS Release 12.3R2, Juniper Networks EX Series switches support port mirroring for Layer 2 bridging traffic.

Layer port mirroring enables you to specify the manner in which incoming and outgoing packets at specified ports are monitored and the manner in which copies of selected packets are forwarded to another destination, where the packets can be analyzed.

MX Series routers and EX Series switches support Layer 2 port mirroring by performing flow monitoring functions by using a class-of-service (CoS) architecture that is in concept similar to, but in particular different from, other routing platforms and switches.

Like the M120 Multiservice Edge Router and M320 Multiservice Edge Router, MX Series routers and EX Series switches support the mirroring of IPv4, IPv6, and VPLS packets simultaneously.

In a Layer 3 environment, MX Series routers and EX Series switches support the mirroring of IPv4 (`family inet`) and IPv6 (`family inet6`) traffic. For information about Layer 3 port mirroring, see the Routing Policies, Firewall Filters, and Traffic Policers User Guide.

Layer 2 Port Mirroring Properties

IN THIS SECTION

- Packet-Selection | **1013**
- Packet Address Family | **1013**
- Mirror Destination Properties | **1014**
- Mirror-Once Option | **1014**

Port mirroring specifies the following types of properties:

Packet-Selection

The packet-selection properties of Layer 2 port-mirroring specify how the sampled packets are to be selected for mirroring:

- The number of packets in each sample.
- The number of packets to mirror from each sample.
- The length to which mirrored packets are to be truncated.

Packet Address Family

The packet address family type specifies the type of traffic to be mirrored. In a Layer 2 environment, MX Series routers and EX Series switches support port mirroring for the following packet address families:

- Family type `ethernet-switching`—For mirroring VPLS traffic when the physical interface is configured with encapsulation type `ethernet-bridge`.
- Family type `ccc`—For mirroring Layer 2 VPN traffic.
- Family type `vpls`—For mirroring VPLS traffic.

NOTE: In typical applications, you send mirrored packets directly to an analyzer, not to another router or switch. If you must send mirrored packets over a network, you should use tunnels. For Layer 2 VPN implementations, you can use the Layer 2 VPN routing instance type `l2vpn` to tunnel the packets to a remote destination.

For information about configuring a routing instance for Layer 2 VPN, see the Junos OS VPNs Library for Routing Devices. For a detailed Layer 2 VPN example configuration, see Junos OS. For information about tunnel interfaces, see the Junos OS Network Interfaces Library for Routing Devices.

Mirror Destination Properties

For a given packet address family, the mirror destination properties of a Layer 2 port-mirroring instance specify how the selected packets are to be sent on a particular physical interface:

- The physical interface on which to send the selected packets.
- Whether filter checking is to be disabled for the mirror destination interface. By default, filter checking is enabled on all interfaces.

> **NOTE**: If you apply a filter to an interface that is also a Layer 2 port-mirroring destination, a commit failure occurs unless you have disabled filter checking for that mirror destination interface.

Mirror-Once Option

If port mirroring is enabled at both ingress and egress interfaces, you can prevent the MX Series router and an EX Series switch from sending duplicate packets to the same destination (which would complicate the analysis of the mirrored traffic).

> **NOTE**: The mirror-once port-mirroring option is a global setting. The option is independent of the packet selection properties and the packet family type-specific mirror destination properties.

Application of Layer 2 Port Mirroring Types

You can apply different sets of Layer 2 port-mirroring properties to the VPLS packets at different ingress or egress points of an MX Series or of an EX Series route.

Table 120 on page 1015 describes the three types of Layer 2 *port mirroring* that you can configure on an MX Series routers and EX Series switches, the: global instance, named instances, and firewall filters.

Table 120: Application of Layer 2 Port Mirroring Types

Type of Layer 2 Port Mirroring Definition	Point of Application	Scope of Mirroring	Description	Configuration Details
Global Instance of Layer 2 Port Mirroring	All ports in the MX Series router (or switch) chassis.	VPLS packets received on all ports in the MX Series router (or switch) chassis.	If configured, the global port-mirroring properties implicitly apply to all the VPLS packets received on all ports in the router (or switch) chassis.	See Configuring the Global Instance of Layer 2 Port Mirroring
Named Instance of Layer 2 Port Mirroring	Ports grouped at the FPC level See "Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level" on page 1136.	VPLS packets received on ports associated with a specific DPC or FPC and its Packet Forwarding Engines.	Overrides any port-mirroring properties configured by the global port-mirroring instance.	See Defining a Named Instance of Layer 2 Port Mirroring. The number of port-mirroring destinations supported for an MX Series router and for an EX Series switch are limited to the number of Packet Forwarding Engines contained on the DPCs or FPCs installed in the router or switch chassis.
	Ports grouped at the PIC level See "Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level" on page 1137.	VPLS packets received on ports associated with a specific Packet Forwarding Engine.	Overrides any port-mirroring properties configured at the FPC level or in the global port-mirroring instance.	

Table 120: Application of Layer 2 Port Mirroring Types *(Continued)*

Type of Layer 2 Port Mirroring Definition	Point of Application	Scope of Mirroring	Description	Configuration Details
Layer 2 Port-Mirroring Firewall Filter	*Logical interface* (including an aggregated Ethernet interface) See Applying Layer 2 Port Mirroring to a Logical Interface.	VPLS packets received or sent on a logical interface.	In the *firewall filter* configuration, include *action* and *action-modifier* terms to apply to the packets selected for mirroring: • The `accept`action is recommended. • The `port-mirror` modifier implicitly references the port-mirroring properties currently bound to the underlying physical interfaces. • The `port-mirror-instance` *`pm-instance-name`* modifier explicitly references a named instance of port mirroring. • (Optional) For tunnel interface input packets only, to mirror the packets to additional destinations, include the `next-hop-group` *`next-hop-group-name`* modifier. This modifier references a next-hop-group that specifies the next-hop addresses (for sending additional copies of packets to an analyzer).	See Defining a Layer 2 Port-Mirroring Firewall Filter. **NOTE**: Layer 2 port-mirroring firewall filters are not supported for logical systems. For mirroring tunnel interface input packets to multiple destinations, also see Defining a Next-Hop Group for Layer 2 Port Mirroring.
	VLAN forwarding table or flood table See "Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain" on page 1169.	Layer 2 traffic forwarded or flooded to a VLAN		
	VPLS routing instance forwarding table or flood table See Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance.	Layer 2 traffic forwarded or flooded to a VPLS routing instance		

Restrictions on Layer 2 Port Mirroring

The following restrictions apply to Layer 2 *port mirroring*:

- Only Layer 2 transit data (packets that contain chunks of data transiting the routing platform or switch as they are forwarded from a source to a destination) can be mirrored. Layer 2 local data (packets that contain chunks of data that are destined for or sent by the Routing Engine, such as Layer 2 control packets) are not mirrored.

- If you apply a port-mirroring filter to the output of a *logical interface*, only Unicast packets are mirrored. To mirror Broadcast packets, Multicast packets, Unicast packets with an unknown destination media access control (MAC) address, or packets with a MAC entry in the destination MAC (DMAC) routing table, apply a filter to the input to the flood table of a VLAN or virtual private LAN service (VPLS) routing instance.

- The mirror destination device should be on a dedicated VLAN and should not participate in any bridging activity; the mirror destination device should not have a bridge to the ultimate traffic destination, and the mirror destination device should not send the mirrored packets back to the source address.

- For either the global port-mirroring instance or a named port-mirroring instance, you can configure only one mirror output interface per port-mirroring instance and packet address family. If you include more than one `interface` statement under the `family (ethernet-switching | ccc | vpls) output` statement, the previous `interface` statement is overridden.

- Layer 2 port-mirroring firewall filtering is not supported for logical systems.

  In a Layer 2 port-mirroring *firewall filter* definition, the *`action-modifier`* filter (`port-mirror` or `port-mirror-instance` *`pm-instance-name`*) relies on port-mirroring properties defined in the global instance or named instances of Layer 2 port mirroring, which are configured under the `[edit forwarding-options port-mirroring]` hierarchy. Therefore, the *`term`* filter cannot support Layer 2 port mirroring for logical systems.

- For a Layer 2 port mirroring firewall filter in which you implicitly reference Layer 2 port mirroring properties by including the `port-mirror` statement, if multiple named instances of Layer 2 port mirroring are bound to the underlying physical interface, then only the first binding in the stanza (or the only binding) is used at the logical interface. This is done for backward compatibility.

- Layer 2 port-mirroring firewall filters do not support the use of next-hop subgroups for load-balancing mirrored traffic.

Configuring Port Mirroring and Analyzers

IN THIS SECTION

- Understanding Port Mirroring Analyzers | **1018**
- Configuring Mirroring on EX9200 Switches to Analyze Traffic (CLI Procedure) | **1026**
- Configuring Mirroring on EX4300 Switches to Analyze Traffic (CLI Procedure) | **1036**
- Configuring Port Mirroring to Analyze Traffic (CLI Procedure) | **1041**
- Verifying Input and Output for Port Mirroring Analyzers on EX Series Switches | **1045**
- Example: Configuring Port Mirroring Analyzers for Local Monitoring of Employee Resource Use | **1047**
- Example: Configuring Port Mirroring for Remote Monitoring of Employee Resource Use | **1053**
- Example: Configuring Mirroring to Multiple Interfaces for Remote Monitoring of Employee Resource Use on EX9200 Switches | **1067**
- Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use Through a Transit Switch on EX9200 Switches | **1079**
- Example: Configuring Mirroring for Local Monitoring of Employee Resource Use on EX4300 Switches | **1089**
- Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use on EX4300 Switches | **1099**
- Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use Through a Transit Switch on EX4300 Switches | **1113**

Understanding Port Mirroring Analyzers

IN THIS SECTION

- Analyzer Overview | **1019**
- Statistical Analyzer Overview | **1020**
- Default Analyzer Overview | **1020**
- Port Mirroring at a Group of Ports Bound to Multiple Statistical Analyzers | **1020**
- Port Mirroring Analyzer Terminology | **1020**
- Configuration Guidelines for Port Mirroring Analyzers | **1023**

Port mirroring can be used for traffic analysis on routers and switches that, unlike hubs, do not broadcast packets to every port on the destination device. Port mirroring sends copies of all packets or policy-based sample packets to local or remote analyzers where you can monitor and analyze the data.

In the context of port mirroring analyzers, we use the term *switching device*. The term indicates that the device (including routers) is performing a switching function.

You can use analyzers on a packet level to help you:

- Monitor network traffic
- Enforce network usage policies
- Enforce file sharing policies
- Identify the causes of problems
- Identify stations or applications with heavy or abnormal bandwidth usage

You can configure port mirroring to mirror:

- Bridged packets (Layer 2 packets)
- Routed packets (Layer 3 packets)

Mirrored packets can be copied to either a local interface for local monitoring or a VLAN or bridge domain for remote monitoring.

The following packets can be copied:

- **Packets entering or exiting a port**—You can mirror packets entering or exiting ports, in any combination, for up to 256 ports. For example, you can send copies of the packets entering some ports and the packets exiting other ports to the same local analyzer port or analyzer VLAN.
- **Packets entering or exiting a VLAN or bridge domain**—You can mirror the packets entering or exiting a VLAN or bridge domain to either a local analyzer port or to an analyzer VLAN or bridge domain. You can configure multiple VLANs (up to 256 VLANs) or bridge domains as ingress inputs to an analyzer, including a VLAN range and private VLANs (PVLANs).
- **Policy-based sample packets**—You can mirror a policy-based sample of packets that are entering a port, VLAN, or bridge domain. You configure a firewall filter with a policy to select the packets to be mirrored. You can send the sample to a port-mirroring instance or to an analyzer VLAN or bridge domain.

Analyzer Overview

You can configure an analyzer to define both the input traffic and the output traffic in the same analyzer configuration. The input traffic to be analyzed can be either traffic that enters or traffic that exits an

interface or VLAN. The analyzer configuration enables you to send this traffic to an output interface, instance, next-hop group, VLAN, or bridge domain. You can configure an analyzer at the `[edit forwarding-options analyzer]` hierarchy level.

Statistical Analyzer Overview

You can define a set of mirroring properties, such as mirroring rate and maximum packet length for traffic, that you can explicitly bind to physical ports on the router or switch. This set of mirroring properties constitutes a statistical analyzer (also called a non-default analyzer). At this level, you can bind a named instance to the physical ports associated with a specific FPC.

Default Analyzer Overview

You can configure an analyzer without configuring any mirroring properties (such as mirroring rate or maximum packet length). By default, the mirroring rate is set to 1 and the maximum packet length is set to the complete length of the packet. These properties are applied at the global level and need not be bound to a specific FPC.

Port Mirroring at a Group of Ports Bound to Multiple Statistical Analyzers

You can apply up to two statistical analyzers to the same port groups on the switching device. By applying two different statistical analyzer instances to the same FPC or Packet Forwarding Engine, you can bind two distinct Layer 2 mirroring specifications to a single port group. Mirroring properties that are bound to an FPC override any analyzer (default analyzer) properties bound at the global level on the switching device. Default analyzer properties are overridden by binding a second analyzer instance on the same port group.

Port Mirroring Analyzer Terminology

Table 121 on page 1021 lists some port mirroring analyzer terms and their descriptions.

Table 121: Analyzer Terminology

Term	Description
Analyzer	In a mirroring configuration, the analyzer includes: • The name of the analyzer • Source (input) ports, VLANs, or bridge domains • The destination for mirrored packets (either a local port, VLAN, or bridge domain)
Analyzer output interface (Also known as a monitor port)	Interface where mirrored traffic is sent and a protocol analyzer is connected. Interfaces used as output to an analyzer must be configured under the `forwarding-options` hierarchy level. Analyzer output interfaces have the following limitations: • They cannot also be a source port. • They do not participate in Layer 2 protocols, such as the Spanning Tree Protocol (STP). • If the bandwidth of the analyzer output interface is not sufficient to handle the traffic from the source ports, overflow packets are dropped.
Analyzer VLAN or bridge domain (Also known as a monitor VLAN or bridge domain)	VLAN or bridge domain to where mirrored traffic is sent to be used by a protocol analyzer. The member interfaces in the monitor VLAN or bridge domain are spread across the switching devices in your network.
Bridge-domain-based analyzer	An analyzer session configured to use bridge domains for input, output or both.
Default analyzer	An analyzer with default mirroring parameters. By default, the mirroring rate is 1 and the maximum packet length is the length of the complete packet.

Table 121: Analyzer Terminology *(Continued)*

Term	Description
Input interface (Also known as mirrored ports or monitored interfaces)	An interface on the switching device where the traffic entering or exiting this interface is mirrored.
LAG-based analyzer	An analyzer that has a link aggregation group (LAG) specified as the input (ingress) interface in the analyzer configuration.
Local mirroring	An analyzer configuration in which packets are mirrored to a local analyzer port.
Monitoring station	A computer running a protocol analyzer.
Analyzer based on next-hop group	An analyzer configuration that uses the next-hop group as the output to an analyzer.
Port-based analyzer	An analyzer configuration that defines interfaces for input and output.
Protocol analyzer application	An application used to examine packets transmitted across a network segment. Also commonly called a network analyzer, packet sniffer or probe.
Remote mirroring	Functions the same way as local mirroring, except that the mirrored traffic is not copied to a local analyzer port but is flooded to an analyzer VLAN or bridge domain that you create specifically for the purpose of receiving mirrored traffic. Mirrored packets have an additional outer tag of the analyzer VLAN or bridge domain.
Statistical analyzer (Also known as a non-default analyzer)	A set of mirroring properties that you can explicitly bind to the physical ports on the switch. This set of analyzer properties is known as a statistical analyzer.
VLAN-based analyzer	An analyzer configuration that uses VLANs to deliver the mirrored traffic to the analyzer.

Configuration Guidelines for Port Mirroring Analyzers

When you configure port mirroring analyzers. we recommend that you follow these guidelines to ensure optimum benefit. We recommend that you disable mirroring when you are not using it, and that you select specific interfaces as input to the analyzer rather than using the `all` keyword option, which enables mirroring on all interfaces. Mirroring only necessary packets reduces any potential performance impact.

You can also limit the amount of mirrored traffic by:

- Using statistical sampling
- Using a firewall filter
- Setting a ratio to select a statistical sample

With local mirroring, traffic from multiple ports is replicated to the analyzer output interface. If the output interface for an analyzer reaches capacity, packets are dropped. You must consider whether the traffic being mirrored exceeds the capacity of the analyzer output interface.

Table 122 on page 1023 summarizes further configuration guidelines for analyzers.

Table 122: Configuration Guidelines for Port Mirroring Analyzers

Guideline	Value or Support Information	Comment
Number of analyzers that you can enable concurrently.	64 Default analyzers 2 per FPC–Statistical analyzer	Statistical analyzers must be bound to an FPC for mirroring traffic on ports belonging to that FPC. **NOTE**: Default analyzer properties are implicitly bound on the last (or second to last) instance on all FPCs in the system. Therefore, when you explicitly bind a second statistical analyzer on the FPC, the default analyzer properties are overridden.
Number of interfaces, VLANs, or bridge domains that you can use as ingress input to an analyzer.	256	–

Table 122: Configuration Guidelines for Port Mirroring Analyzers *(Continued)*

Guideline	Value or Support Information	Comment
Types of ports on which you cannot mirror traffic.	• *Virtual Chassis* ports (VCPs) • Management Ethernet ports (me0 or vme0) • Integrated routing and bridging (IRB) interfaces • VLAN-tagged Layer 3 interfaces	
Protocol families that you can include in an analyzer.	`ethernet-switching` for EX Series switches and `bridge` for MX Series routers.	An analyzer mirrors only bridged traffic. To mirror routed traffic, use the port mirroring configuration with `family` as `inet` or `inet6`.
Packets with physical layer errors are not sent to the local or remote analyzer.	Applicable	Packets with these errors are filtered out and thus are not sent to the analyzer.
Analyzer does not support line-rate traffic.	Applicable	Mirroring for line-rate traffic is done on a best-effort basis.
Analyzer output on a LAG interface.	Supported	

Table 122: Configuration Guidelines for Port Mirroring Analyzers *(Continued)*

Guideline	Value or Support Information	Comment
Analyzer output interface mode as trunk mode.	Supported	• The trunk interface has to be a member of all VLANs or bridge domains that are related to the input configuration of the analyzer. • You must use the `mirror-once` option if the input has been configured as VLAN or bridge domain and the output is a trunk interface. **NOTE**: With the mirror-once option, if the analyzer input is from both ingress and egress mirroring, only ingress traffic is mirrored. If both ingress and egress mirroring are required, the output interface cannot be a trunk. In such cases, configure the interface as an access interface.
Egress mirroring of host-generated control packets.	Not supported	
Configuring Layer 3 logical interfaces in the `input` stanza of an analyzer.	Not supported	
The analyzer input and output stanzas containing members of the same VLAN or the VLAN itself must be avoided.	Applicable	
Support for VLAN and its member interfaces in different analyzer sessions	Not supported	If mirroring is configured, either of the analyzers is active.

Table 122: Configuration Guidelines for Port Mirroring Analyzers *(Continued)*

Guideline	Value or Support Information	Comment
Egress mirroring of aggregated Ethernet (ae) interfaces and its child logical interfaces configured for different analyzers.	Not supported	

Configuring Mirroring on EX9200 Switches to Analyze Traffic (CLI Procedure)

IN THIS SECTION

- Configuring an Analyzer for Local Traffic Analysis | **1027**
- Configuring an Analyzer for Remote Traffic Analysis | **1028**
- Configuring a Statistical Analyzer for Local Traffic Analysis | **1029**
- Configuring a Statistical Analyzer for Remote Traffic Analysis | **1030**
- Binding Statistical Analyzers to Ports Grouped at the FPC Level | **1032**
- Configuring an Analyzer with Multiple Destinations by Using Next-Hop Groups | **1034**
- Defining a Next-Hop Group for Layer 2 Mirroring | **1035**

EX9200 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy the following packets:

- Packets entering or exiting a port
- Packets entering or exiting a VLAN

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable the analyzers that you have configured when you are not using them.

- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by:
  - Using statistical sampling.
  - Setting ratios to select statistical samples.
  - Using firewall filters.

NOTE: If you want to create additional analyzers without deleting the existing analyzers, disable the existing analyzers by using the `disable analyzer` *`analyzer-name`* statement from the command-line-interface (CLI) or from the J-Web configuration page for mirroring.

NOTE: Interfaces used as output to an analyzer must be configured under the `ethernet-switching family`, and must be associated to a VLAN.

Configuring an Analyzer for Local Traffic Analysis

To mirror network traffic or VLAN traffic on the switch to an interface on the switch by using analyzers:

1. Choose a name for the analyzer and specify the input:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name
```

For example, create an analyzer called `employee-monitor` to monitor the packets entering interfaces ge-0/0/0.0 and ge-0/0/1.0:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0

[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

2. Configure the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output interface interface-name
```

For example, configure ge-0/0/10.0 as the destination interface for the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output interface ge-0/0/10.0
```

Configuring an Analyzer for Remote Traffic Analysis

To mirror traffic that is traversing interfaces or a VLAN on the switch to a VLAN used for analysis from a remote location:

1. Configure a VLAN to carry the mirrored traffic:

```
[edit]
user@switch# set vlans analyzer-name vlan-id vlan-ID
```

For example, define an analyzer VLAN called `remote-analyzer` and assign it the VLAN ID `999`:

```
[edit]
user@switch# set vlans remote-analyzer vlan-id 999
```

2. Set the interface that is connected to the distribution switch to access mode and associate it with the analyzer VLAN:

```
[edit]
user@switch# set interfaces interface-name unit 0 family ethernet-switching interface-mode
access vlan members vlan-ID
```

For example, set the interface ge-0/1/1 to access mode and associate it with the analyzer VLAN ID `999`:

```
[edit]
user@switch# set interfaces ge-0/1/1 unit 0 family ethernet-switching interface-mode access
vlan members 999
```

3. Configure the analyzer:

   a. Define an analyzer and specify the traffic to be mirrored:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name
```

   For example, define the `employee-monitor` analyzer for which traffic to be mirrored comprises packets entering interfaces ge-0/0/0.0 and ge-0/0/1.0:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

   b. Specify the analyzer VLAN as the output for the analyzer:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output vlan vlan-ID
```

   For example, specify the `remote-analyzer` VLAN as the output analyzer for the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output vlan 999
```

Configuring a Statistical Analyzer for Local Traffic Analysis

To mirror interface traffic or VLAN traffic on the switch to an interface on the switch by using a statistical analyzer:

1. Choose a name for the analyzer and specify the input interfaces:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name

[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

For example, specify an analyzer called `employee-monitor` and specify the input interfaces ge-0/0/0 and ge-0/0/1:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0

[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

2. Configure the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output interface interface-name
```

For example, configure ge-0/0/10.0 as the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output interface ge-0/0/10.0
```

3. Specify mirroring properties.
   a. Specify the mirroring rate—that is, the number of packets to be mirrored per second:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input rate number
```

The valid range is 1 through 65,535.

   b. Specify at what length mirrored packets are truncated:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input maximum-packet-length number
```

The valid range is 0 through 9216. The default value is 0, indicating that mirrored packets are not truncated.

Configuring a Statistical Analyzer for Remote Traffic Analysis

To mirror traffic that is traversing interfaces or a VLAN on the switch to a VLAN for analysis from a remote location by using a statistical analyzer:

1. Configure a VLAN to carry the mirrored traffic:

```
[edit]
user@switch# set vlans vlan-name vlan-id vlan-ID
```

   For example, configure a VLAN called `remote-analyzer` with VLAN ID `999`:

```
[edit]
user@switch# set vlans remote-analyzer vlan-id 999
```

2. Set the interface that is connected to the distribution switch to access mode and associate it with the VLAN:

```
[edit]
user@switch# set interfaces interface-name unit 0 family ethernet-switching interface-mode
access vlan members vlan-ID
```

   For example, set the interface ge-0/1/1.0 that is connected to the distribution switch to access mode and associate it with the `remote-analyzer` VLAN:

```
[edit]
user@switch# set interfaces ge-0/1/1.0 unit 0 family ethernet-switching interface-mode access
vlan members 999
```

3. Configure the statistical analyzer:

   a. Specify the traffic to be mirrored:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name
```

      For example, specify the packets entering ports ge-0/0/0.0 and ge-0/0/1.0 to be mirrored:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

b. Specify an output for the analyzer:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output vlan vlan-ID
```

For example, specify the `remote-analyzer` VLAN as the output for the analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output vlan 999
```

4. Specify mirroring properties.

   a. Specify the mirroring rate—that is, the number of packets to be mirrored per second:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input rate number
```

   The valid range is 1 through 65,535.

   b. Specify the length to which mirrored packets are to be truncated:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input  maximum-packet-length number
```

   The valid range is 0 through 9216. The default value is 0, which means the mirrored packets are not truncated.

Binding Statistical Analyzers to Ports Grouped at the FPC Level

You can bind a statistical analyzer to a specific FPC in the switch, that is, you can bind the statistical analyzer instance at the FPC level of the switch. The mirroring properties specified in the statistical analyzer are applied to all physical ports associated with all Packet Forwarding Engines on the specified FPC.

To bind a named instance of Layer 2 analyzer to an FPC:

1. Enable configuration of switch chassis properties:

```
[edit]
user@switch# edit chassis
```

2. Enable configuration of an FPC (and its installed PICs):

```
[edit chassis]
user@switch# edit fpc slot-number
```

3. Bind a statistical analyzer instance to the FPC:

```
[edit chassis fpc slot-number]
user@switch# set port-mirror-instance stats_analyzer-1
```

4. (Optional) To bind a second statistical analyzer instance of Layer 2 mirroring to the same FPC, repeat Step 3 and specify a different statistical analyzer name:

```
[edit chassis fpc slot-number]
user@switch# set port-mirror-instance stats_analyzer-2
```

5. Verify the minimum configuration of the binding:

```
[edit chassis fpc slot-number port-mirror-instance analyzer_name]
user@switch# top
[edit]
user@switch# show chassis
chassis {
   fpc slot-number { # Bind two statistical analyzers or port mirroring
                       named instances at the FPC level.
   port-mirror-instance stats_analyzer-1;
   port-mirror-instance stats_analyzer-2;
                   }
               }
```

NOTE: On binding a second instance (`stats_analyzer-2` in this example), the mirroring properties of this session, if configured, overrides any default analyzer.

Configuring an Analyzer with Multiple Destinations by Using Next-Hop Groups

You can mirror traffic to multiple destinations by configuring next-hop groups as analyzer output. The mirroring of packets to multiple destinations is also known as multipacket port mirroring.

To mirror interface traffic or VLAN traffic on the switch to an interface on the switch (by using analyzers):

1. Choose a name for the analyzer and specify the input:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name
```

   For example, create an analyzer called `employee-monitor` for which the input traffic comprises packets entering interfaces ge-0/0/0.0 and ge-0/0/1.0:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

2. Configure the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output next-hop-group next-hop-group-name
```

   For example, configure the next-hop group `nhg` as the destination for the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output next-hop-group nhg
```

Defining a Next-Hop Group for Layer 2 Mirroring

The next-hop group configuration at the `[edit forwarding-options]` configuration level enables you to define a next-hop group name, the type of addresses to be used in the next-hop group, and the logical interfaces that form the multiple destinations to which traffic can be mirrored. By default, the next-hop group is specified using Layer 3 addresses using the `[edit forwarding-options next-hop-group` *`next-hop-group-name`* `group-type inet]` statement. To specify a next-hop group using Layer 2 addresses instead, include the `[edit forwarding-options next-hop-group` *`next-hop-group-name`* `group-type layer-2]` statement.

To define a next-hop group for Layer 2 mirroring:

1. Enable configuration of a next-hop group for Layer 2 mirroring:

   ```
   [edit forwarding-options ]
   user@switch# set next-hop-group next-hop-group-name
   ```

   For example, configure `next-hop-group` with name `nhg`:

   ```
   [edit forwarding-options]
   user@switch# set next-hop-group nhg
   ```

2. Specify the type of addresses to be used in the next-hop group configuration:

   ```
   [edit forwarding-options next-hop-group next-hop-group-name]
   user@switch# set group-type layer-2
   ```

   For example, configure `next-hop-group type` as `layer-2` because the analyzer output must be `layer-2` only:

   ```
   [edit forwarding-options]
   user@switch# set next-hop-group nhg group-type layer-2
   ```

3. Specify the logical interfaces of the next-hop group:

   ```
   [edit forwarding-options next-hop-group next-hop-group-name]
   user@switch# set interface logical-interface-name-1
   user@switch# set interface logical-interface-name-2
   ```

For example, to specify ge-0/0/10.0 and ge-0/0/11.0 as the logical interfaces of the next-hop group nhg:

```
[edit forwarding-options]
user@switch# set next-hop-group nhg interface ge-0/0/10.0
user@switch# set next-hop-group nhg interface ge-0/0/11.0
```

Configuring Mirroring on EX4300 Switches to Analyze Traffic (CLI Procedure)

IN THIS SECTION

- Configuring an Analyzer for Local Traffic Analysis | **1037**
- Configuring an Analyzer for Remote Traffic Analysis | **1038**
- Configuring Port Mirroring | **1039**

NOTE: This task uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style.

EX4300 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering a VLAN

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured mirroring configurations when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by using firewall filters.

NOTE: If you want to create additional analyzers without deleting the existing analyzers, then disable the existing analyzers by using the `disable analyzer` *`analyzer-name`* statement from the command-line interface or the J-Web configuration page for mirroring.

NOTE: Interfaces used as output for an analyzer must be configured under the `ethernet-switching` family.

Configuring an Analyzer for Local Traffic Analysis

To mirror interface traffic or VLAN traffic on the switch to an interface on the switch (by using analyzers):

1. Choose a name for the analyzer and specify the input:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name
```

   For example, create an analyzer called `employee-monitor` for which the input traffic is packets entering interfaces ge-0/0/0.0 and ge-0/0/1.0:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0

[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

2. Configure the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output interface interface-name
```

   For example, configure ge-0/0/10.0 as the destination interface for the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output interface ge-0/0/10.0
```

Configuring an Analyzer for Remote Traffic Analysis

To mirror traffic that is traversing interfaces or a VLAN on the switch to a VLAN for analysis from a remote location (by using analyzers):

1. Configure a VLAN to carry the mirrored traffic:

```
[edit]
user@switch# set vlans analyzer-name vlan-id vlan-ID
```

   For example, define an analyzer VLAN called `remote-analyzer` and assign it a VLAN ID of `999`:

```
[edit]
user@switch# set vlans remote-analyzer vlan-id 999
```

2. Set the uplink module interface that is connected to the distribution switch to trunk mode and associate it with the analyzer VLAN:

```
[edit]
user@switch# set interfaces interface-name unit 0 family ethernet-switching interface-mode
trunk vlan members vlan-ID
```

   For example, set the interface ge-0/1/1 to trunk mode and associate it with the analyzer VLAN ID `999`:

```
[edit]
user@switch# set interfaces ge-0/1/1 unit 0 family ethernet-switching interface-mode trunk
vlan members 999
```

3. Configure the analyzer:

   a. Define an analyzer and specify the traffic to be mirrored:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress interface interface-name
```

For example, define the `employee-monitor` analyzer for which traffic to be mirrored is packets entering interfaces ge-0/0/0.0 and ge-0/0/1.0:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

b. Specify the analyzer VLAN as the output for the analyzer:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output vlan vlan-ID
```

For example, specify the `remote-analyzer` VLAN as the output analyzer for the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output vlan 999
```

Configuring Port Mirroring

To filter packets to be mirrored to a port-mirroring instance, create the instance and then use it as the action in the firewall filter. You can use firewall filters in both local and remote mirroring configurations.

If the same port-mirroring instance is used in multiple filters or terms, the packets are copied to the analyzer output port or analyzer VLAN only once.

To filter mirrored traffic, create a port-mirroring instance under the `[edit forwarding-options]` hierarchy level, and then create a firewall filter. The filter can use any of the available match conditions and must have `port-mirror-instance` *`instance-name`* as an action. This action in the firewall filter configuration provides the input to the port-mirroring instance.

To configure a port-mirroring instance with firewall filters:

1. Configure the port-mirroring instance name (here, `employee-monitor`) and the output:

a. For local analysis, set the output to the local interface where you will connect the computer running the protocol analyzer:

```
[edit forwarding-options]
user@switch# set port-mirroring instance employee-monitor output interface ge-0/0/10.0
```

b. For remote analysis, set the output to the `remote-analyzer` VLAN:

```
[edit forwarding-options]
user@switch# set port-mirroring instance employee-monitor output vlan 999
```

2. Create a firewall filter by using any of the available match conditions and assign `employee-monitor` to the `port-mirror-instance` action:

   This step shows a firewall filter `example-filter`, with two terms (`no-analyzer` and `to-analyzer`):

   a. Create the first term to define the traffic that should not pass through to the port-mirroring instance `employee-monitor`:

```
[edit firewall family ethernet-switching
user@switch# set filter example-filter term no-analyzer from source-address ip-address
user@switch# set filter example-filter term no-analyzer from destination-address ip-address
user@switch# set filter example-filter term no-analyzer then accept
```

   b. Create the second term to define the traffic that should pass through to the port-mirroring instance `employee-monitor`:

```
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer from destination-port 80
user@switch# set filter example-filter term to-analyzer then port-mirror-instance employee-
monitor
user@switch# set filter example-filter term to-analyzer then accept
```

3. Apply the firewall filter to the interfaces or VLAN that provide input to the port-mirroring instance:

```
[edit]
user@switch# set interfaces ge-0/0/0 unit 0 family ethernet-switching filter input example-
filter
ser@switch# set vlan remote-analyzer filter input example-filter
```

Configuring Port Mirroring to Analyze Traffic (CLI Procedure)

IN THIS SECTION

- Configuring Port Mirroring for Local Traffic Analysis | **1042**
- Configuring Port Mirroring for Remote Traffic Analysis | **1043**
- Filtering the Traffic Entering an Analyzer | **1044**

This configuration task uses Junos OS for EX Series switches that do not support the Enhanced Layer 2 Software (ELS) configuration style.

EX Series switches allow you to configure port mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use port mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering a VLAN on EX2200, EX3200, EX3300, EX4200, EX4500, or EX6200 switches
- Packets exiting a VLAN on EX8200 switches

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured port mirroring analyzers when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by:
  - Using statistical sampling.
  - Setting ratios to select statistical samples.
  - Using firewall filters.

Before you begin to configure port mirroring, note the following limitations for analyzer output interfaces:

- Cannot also be a source port.

- Cannot be used for switching.
- Do not participate in Layer 2 protocols (such as RSTP) when part of a port mirroring configuration.
- Do not retain any VLAN associations they held before they were configured as analyzer output interfaces.

NOTE: If you want to create additional analyzers without deleting the existing analyzer, first disable the existing analyzer using the `disable analyzer` *`analyzer-name`* command or the J-Web configuration page for port mirroring.

NOTE: Interfaces used as output for an analyzer must be configured as family `ethernet-switching`.

Configuring Port Mirroring for Local Traffic Analysis

To mirror interface traffic or VLAN traffic on the switch to another interface on the switch:

1. Choose a name for the analyzer—in this case `employee-monitor`—and specify the input—in this case, packets entering `ge-0/0/0` and `ge-0/0/1`:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor ingress interface ge-0/0/0.0

[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

2. Optionally, you can specify a statistical sampling of the packets by setting a ratio:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor ratio 200
```

When the ratio is set to 200, 1 of every 200 packets is mirrored to the analyzer. You can use statistical sampling to reduce the volume of mirrored traffic, as a high volume of mirrored traffic can be performance intensive for the switch. On EX8200 switches, you can set a ratio only for ingress packets.

3. Configure the destination interface for the mirrored packets:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor output interface ge-0/0/10.0
```

Configuring Port Mirroring for Remote Traffic Analysis

To mirror traffic that is traversing interfaces or a VLAN on the switch to a VLAN for analysis from a remote location:

1. Configure a VLAN to carry the mirrored traffic. This VLAN is called `remote-analyzer` and given the ID of 999 by convention in this documentation:

```
[edit]
user@switch# set vlans remote-analyzer vlan-id 999
```

2. Set the uplink module interface that is connected to the distribution switch to trunk mode and associate it with the `remote-analyzer` VLAN:

```
[edit]
user@switch# set interfaces ge-0/1/1 unit 0 family ethernet-switching port-mode trunk vlan
members 999
```

3. Configure the analyzer:

   a. Choose a name and set the loss priority to high. Loss priority should always be set to high when configuring for remote port mirroring:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor loss-priority high
```

   b. Specify the traffic to be mirrored—in this example the packets entering ports `ge-0/0/0` and `ge-0/0/1`:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

c. Specify the `remote-analyzer` VLAN as the output for the analyzer:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor output vlan 999
```

4. Optionally, you can specify a statistical sampling of the packets by setting a ratio:

```
[edit ethernet-switching-options]
user@switch# set analyzer employee-monitor ratio 200
```

When the ratio is set to 200, 1 out of every 200 packets is mirrored to the analyzer. You can use this to reduce the volume of mirrored traffic as a very high volume of mirrored traffic can be performance intensive for the switch.

Filtering the Traffic Entering an Analyzer

To filter which packets are mirrored to an analyzer, create the analyzer and then use it as the action in the firewall filter. You can use firewall filters in both local and remote port mirroring configurations.

If the same analyzer is used in multiple filters or terms, the packets are copied to the analyzer output port or analyzer VLAN only once.

To filter mirrored traffic, create an analyzer and then create a firewall filter. The filter can use any of the available match conditions and must have an action of `analyzer`. The action of the firewall filter provides the input to the analyzer.

To configure port mirroring with filters:

1. Configure the analyzer name (here, `employee-monitor`) and the output:

   a. For local analysis, set the output to the local interface to which you will connect the computer running the protocol analyzer application:

   ```
   [edit ethernet-switching-options]
   user@switch# set analyzer employee-monitor output interface ge-0/0/10.0
   ```

   b. For remote analysis, set the loss priority to high and set the output to the `remote-analyzer` VLAN:

   ```
   [edit ethernet-switching-options]
   user@switch# set analyzer employee-monitor loss-priority high output vlan 999
   ```

2. Create a firewall filter using any of the available match conditions and specify the action as `analyzer`:

This step shows a firewall filter called `example-filter`, with two terms:

a. Create the first term to define the traffic that should not pass through to the analyzer:

```
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term no-analyzer from source-address ip-address
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term no-analyzer from destination-address ip-address
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term no-analyzer then accept
```

b. Create the second term to define the traffic that should pass through to the analyzer:

```
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer from destination-port 80
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer then analyzer employee-monitor
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer then accept
```

3. Apply the firewall filter to the interfaces or VLAN that are input to the analyzer:

```
[edit]
user@switch# set interfaces ge-0/0/0 unit 0 family ethernet-switching filter input example-
filter

[edit]
user@switch# set vlan remote-analyzer filter input example-filter
```

Verifying Input and Output for Port Mirroring Analyzers on EX Series Switches

IN THIS SECTION

- Purpose | 1046
- Action | 1046
- Meaning | 1047

Purpose

This verification task uses Junos OS for EX Series switches that do not support the Enhanced Layer 2 Software (ELS) configuration style.

Verify that an analyzer has been created on the switch and has the appropriate mirror input interfaces, and the appropriate analyzer output interface.

Action

You can verify the port mirror analyzer is configured as expected by using the `show analyzer` command.

```
[edit]
user@switch>                        show analyzer
Analyzer name                  : employee-monitor
  Output VLAN                  : remote-analyzer
  Mirror ratio                 : 1
  Loss priority                : High
  Ingress monitored interfaces : ge-0/0/0.0
  Ingress monitored interfaces : ge-0/0/1.0
```

You can view all of the port mirror analyzers configured on the switch, including any that are disabled, by using the `show ethernet-switching-options` command in configuration mode.

```
user@switch# show ethernet-switching-options
inactive: analyzer employee-web-monitor {
    loss-priority high;
    output {

analyzer employee-monitor {
    loss-priority high;
    input {
        ingress {
            interface ge-0/0/0.0;
            interface ge-0/0/1.0;
        }
    }
    output {
        vlan {
            remote-analyzer;
        }
```

```
    }
}
```

Meaning

This output shows that the employee-monitor analyzer has a ratio of 1 (mirroring every packet, the default), a loss priority of `high` (set this option to `high` whenever the analyzer output is to a VLAN), is mirroring the traffic entering ge-0/0/0 and ge-0/0/1, and is sending the mirrored traffic to the analyzer called remote-analyzer.

Example: Configuring Port Mirroring Analyzers for Local Monitoring of Employee Resource Use

IN THIS SECTION

- Requirements | **1047**
- Overview and Topology | **1048**
- Mirroring All Employee Traffic for Local Analysis | **1049**
- Verification | **1052**

Juniper Networks devices allow you to configure port mirroring to send copies of packets to either a local interface for local monitoring, to a VLAN or to a bridge domain for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering or exiting a VLAN or bridge domain

You can then analyze the mirrored traffic locally or remotely using a protocol analyzer. You can install an analyzer on a local destination interface. If you are sending mirrored traffic to an analyzer VLAN or bridge domain, you can use an analyzer on a remote monitoring station.

This topic describes how to configure local mirroring on a switching device. The examples in this topic describe how to configure a switching device to mirror traffic entering interfaces connected to employee computers to an analyzer output interface on that same device.

Requirements

Use either one of the following hardware and software components:

- One EX9200 switch with Junos OS Release 13.2 or later
- One MX Series router with Junos OS Release 14.1 or later

Before you configure port mirroring, be sure you have an understanding of mirroring concepts. For information about analyzers, see "Understanding Port Mirroring Analyzers" on page 1018. For information about port mirroring, see "Understanding Layer 2 Port Mirroring" on page 1012.

Overview and Topology

This topic describes how to mirror all traffic entering ports on the switching device to a destination interface on the same device (local mirroring). In this case, the traffic is entering ports connected to employee computers.

> **NOTE**: Mirroring all traffic requires significant bandwidth and should only be done during an active investigation.

The interfaces ge-0/0/0 and ge-0/0/1 serve as connections for employee computers.

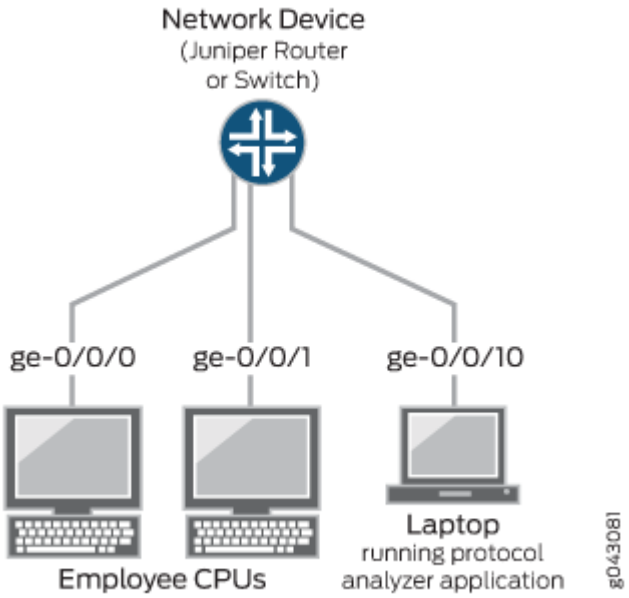
The interface ge-0/0/10 is reserved for analysis of the mirrored traffic.

Connect a PC running a protocol analyzer to the analyzer output interface.

> **NOTE**: Multiple ports mirrored to one interface can cause buffer overflow, resulting in mirrored packets being dropped at the output interface.

Figure 34 on page 1049 shows the network topology for this example.

Figure 34: Network Topology for Local Port Mirroring Example



Mirroring All Employee Traffic for Local Analysis

IN THIS SECTION

- Procedure | 1049

Procedure

CLI Quick Configuration

To quickly configure local mirroring for ingress traffic sent on two ports connected to employee computers, copy either of the following commands for EX Series switches or for MX Series routers and paste them into the switching device terminal window:

EX Series

```
[edit]
set interfaces ge-0/0/0 unit 0 family ethernet-switching
```

```
set interfaces ge-0/0/1 unit 0 family ethernet-switching
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output interface ge-0/0/10.0
```

MX Series

```
[edit]
set interfaces ge-0/0/0 unit 0 family bridge interface-mode access vlan-id 99
set interfaces ge-0/0/1 unit 0 family bridge interface-mode access vlan-id 98
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output interface ge-0/0/10.0
```

Step-by-Step Procedure

To configure an analyzer called `employee-monitor` and specify both the input (source) interfaces and the analyzer output interface:

1. Configure each interface to be used in the analyzer configuration. Use the family protocol that is correct for your platform.

   ```
   EX Series
   [edit]
   set interfaces ge-0/0/0 unit 0 family ethernet-switching
   set interfaces ge-0/0/1 unit 0 family ethernet-switching
   ```

   To configure `family bridge` on an interface, you must configure `interface-mode access` or `interface-mode trunk` as well. You also must configure `vlan-id`.

   ```
   MX Series
   [edit]
   set interfaces ge-0/0/0 unit 0 family bridge interface-mode access vlan-id 99
   set interfaces ge-0/0/1 unit 0 family bridge interface-mode access vlan-id 98
   ```

2. Configure each interface connected to employee computers as an output analyzer interface `employee-monitor`.

```
[edit forwarding-options]
set analyzer employee-monitor input ingress interface ge-0/0/0.0
set analyzer employee-monitor input ingress interface ge-0/0/1.0
```

3. Configure the output analyzer interface for the `employee-monitor` analyzer.

   This will be the destination interface for the mirrored packets.

```
[edit forwarding-options]
set analyzer employee-monitor output interface ge-0/0/10.0
```

Results

Check the results of the configuration.

```
[edit]
user@device# show forwarding-options
analyzer {
    employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
        }
        output {
            interface ge-0/0/10.0;
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | 1052

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer `employee-monitor` has been created on the switching device with the appropriate input interfaces and the appropriate output interface.

Action

Use the `show forwarding-options analyzer` operational command to verify that an analyzer is configured as expected.

```
user@device> show forwarding-options analyzer
Analyzer name                          : employee-monitor
  Mirror rate                          : 1
  Maximum packet length                : 0
  State                                   : up
  Ingress monitored interfaces         : ge-0/0/0.0
  Output interface                        : ge-0/0/10.0
```

Meaning

The output shows that the `employee-monitor` analyzer has a ratio of 1 (that is, mirroring every packet, the default setting), the maximum size of the original packet mirrored is 0 (indicating that the entire packet is mirrored), the state of the configuration is `up`, and the analyzer is mirroring the traffic entering the ge-0/0/0 interface, and sending the mirrored traffic to the ge-0/0/10 interface.

If the state of the output interface is `down` or if the output interface is not configured, the value of `State` will be `down` indicating that the analyzer will not be receiving mirrored traffic.

Example: Configuring Port Mirroring for Remote Monitoring of Employee Resource Use

IN THIS SECTION

- Requirements | 1054
- Overview and Topology | 1054
- Mirroring Employee Traffic for Remote Analysis By Using a Statistical Analyzer | 1055
- Verification | 1066

Juniper Networks devices allow you to configure port mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN or bridge domain for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering or exiting a VLAN
- Packets entering or exiting a bridge domain

If you are sending mirrored traffic to an analyzer VLAN or bridge domain, you can analyze the mirrored traffic by using a protocol analyzer running on a remote monitoring station.

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you do the following:

- Disable your configured mirroring sessions when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by:
  - Using statistical sampling.
  - Setting ratios to select statistical samples.
  - Using firewall filters.

The examples in this topic describe how to configure remote port mirroring to analyze employee resource usage.

Requirements

This example uses one of the following pairs of hardware and software components:

- One EX9200 switch connected to another EX9200 switch, both running Junos OS Release 13.2 or later
- One MX Series router connected to another MX Series router, both running Junos OS Release 14.1 or later

Before you configure remote mirroring, be sure that:

- You have an understanding of mirroring concepts. For information about analyzers, see "Understanding Port Mirroring Analyzers" on page 1018. For information about port mirroring, see "Understanding Layer 2 Port Mirroring" on page 1012.
- The interfaces that the analyzer will use as input interfaces have already been configured on the switching device.

Overview and Topology

IN THIS SECTION

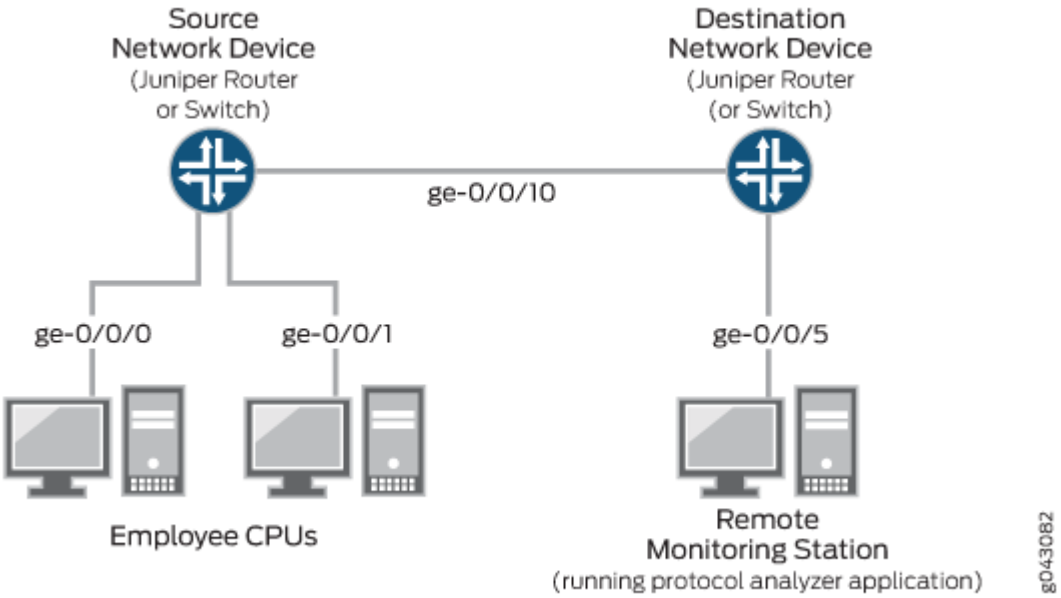
- Topology | **1055**

This topic describes how to configure port mirroring to a remote analyzer VLAN or bridge domain so that the analysis can be done from a remote monitoring station.

Figure 35 on page 1055 shows the network topology for both the EX Series example and the MX Series example scenarios.

Topology

Figure 35: Network Topology for Remote Port Mirroring and Analysis



In this example:

- Interface ge-0/0/0 is a Layer 2 interface, and interface ge-0/0/1 is a Layer 3 interface (both are interfaces on the source device) that serve as connections for employee computers.
- Interface ge-0/0/10 is a Layer 2 interface that connects the source switching device to the destination switching device.
- Interface ge-0/0/5 is a Layer 2 interface that connects the destination switching device to the remote monitoring station.
- The analyzer `remote-analyzer` is configured on all switching devices in the topology to carry the mirrored traffic. This topology can use either a VLAN or a bridge domain.

Mirroring Employee Traffic for Remote Analysis By Using a Statistical Analyzer

IN THIS SECTION

Mirroring Employee Traffic for Remote Analysis for EX Series Switches | **1056**

- Mirroring Employee Traffic for Remote Analysis for MX Series Routers | **1061**

To configure a statistical analyzer for remote traffic analysis for all incoming and outgoing employee traffic, select one of the following examples:

Mirroring Employee Traffic for Remote Analysis for EX Series Switches

CLI Quick Configuration

To quickly configure a statistical analyzer for remote traffic analysis of the incoming and outgoing employee traffic, copy the following commands for EX Series switches and paste them into the correct switching device terminal window.

- Copy and paste the following commands in the *source* switching device terminal window:

  EX Series

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output vlan remote-analyzer
set forwarding-options analyzer employee-monitor input rate 2
set forwarding-options analyzer employee-monitor input maximum-packet-length 128
set chassis fpc 0 port-mirror-instance employee-monitor
```

- Copy and paste the following commands in the *destination* switching device terminal window:

  EX Series

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
set interfaces ge-0/0/5 unit 0 family ethernet-switching interface-mode access
```

```
set forwarding-options analyzer employee-monitor input ingress vlan remote-analyzer
set forwarding-options analyzer employee-monitor output interface ge-0/0/5.0
```

Step-by-Step Procedure

To configure basic remote mirroring:

1. On the source switching device, do the following:

   - Configure the VLAN ID for the `remote-analyzer` VLAN.

     ```
     [edit]
     user@device# set vlans remote-analyzer vlan-id 999
     ```

   - Configure the interface on the network port connected to the destination switching device for access mode and associate it with the `remote-analyzer` VLAN.

     ```
     [edit]
     user@device# set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode
     access
     user@device# set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
     ```

   - Configure the statistical analyzer `employee-monitor`.

     ```
     [edit forwarding-options]
     user@device# set analyzer employee-monitor input ingress interface ge-0/0/0.0
     user@device# set analyzer employee-monitor input ingress interface ge-0/0/1.0
     user@device# set analyzer employee-monitor input egress interface ge-0/0/0.0
     user@device# set analyzer employee-monitor input egress interface ge-0/0/1.0
     user@device# set analyzer employee-monitor output vlan remote-analyzer
     user@device# set analyzer employee-monitor input rate 2
     user@device# set analyzer employee-monitor input maximum-packet-length 128
     ```

   - Bind the statistical analyzer to the FPC that contains the input interface.

     ```
     [edit]
     user@device# set chassis fpc 0 port-mirror-instance employee-monitor
     ```

2. On the destination network device, do the following:

- Configure the VLAN ID for the `remote-analyzer` VLAN.

```
[edit]
user@device# set vlans remote-analyzer vlan-id 999
```

- Configure the interface on the destination switching device for access mode and associate it with the `remote-analyzer` VLAN.

```
[edit interfaces]
user@device# set ge-0/0/10 unit 0 family ethernet-switching interface-mode access
user@device# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the interface connected to the destination switching device for access mode.

```
[edit interfaces]
user@device# set ge-0/0/5 unit 0 family ethernet-switching interface-mode access
```

- Configure the `employee-monitor` analyzer.

```
[edit forwarding-options]
user@device# set analyzer employee-monitor input ingress vlan remote-analyzer
user@device# set analyzer employee-monitor output interface ge-0/0/5.0
```

- Specify mirroring parameters such as rate and the maximum packet length for the `employee-monitor` analyzer.

```
[edit]
user@device# set forwarding-options analyzer employee-monitor input rate 2
user@device# set forwarding-options analyzer employee-monitor input maximum-packet-length
128
```

- Bind the `employee-monitor` analyzer to the FPC containing the input ports.

```
[edit]
user@device# set chassis fpc 0 port-mirror-instance employee-monitor
```

Results

Check the results of the configuration on the source switching device:

```
[edit]
user@device# show
 forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
            egress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }

            maximum-packet-length 128;
            rate 2;
        }
        output {
            vlan {
                remote-analyzer;
            }
        }
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
                vlan {
                    members 999;
                }
            }
        }
    }
}
vlans {
    remote-analyzer {
```

```
        vlan-id 999;
    }
}
```

Check the results of the configuration on the destination switching device.

```
[edit]
user@device# show
interfaces {
    ge0/0/5 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
                vlan {
                    members 999;
                }
            }
        }
    }
}
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/10.0;
        }
    }
}
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                vlan remote-analyzer;
            }
        }
```

```
        output {
            interface {
                ge-0/0/5.0;
            }
        }
    }
}
```

Mirroring Employee Traffic for Remote Analysis for MX Series Routers

CLI Quick Configuration

To quickly configure a statistical analyzer for remote traffic analysis of incoming and outgoing employee traffic, copy the following commands for MX Series routers and paste them into the correct switching device terminal window.

- Copy and paste the following commands in the *source* switching device terminal window:

  MX Series

```
[edit]
set bridge-domains remote-analyzer vlan-id 999
set interfaces ge-0/0/10 unit 0 family bridge interface-mode access
set interfaces ge-0/0/10 unit 0 family bridge vlan-id 999
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output bridge-domain remote-analyzer
set forwarding-options analyzer employee-monitor input rate 2
set forwarding-options analyzer employee-monitor input maximum-packet-length 128
set chassis fpc 0 port-mirror-instance employee-monitor
```

- Copy and paste the following commands in the *destination* switching device terminal window:

  MX Series

```
[edit]
set bridge-domains remote-analyzer vlan-id 999
set interfaces ge-0/0/10 unit 0 family bridge interface-mode access
set interfaces ge-0/0/10 unit 0 family bridge vlan-id 999
```

```
set interfaces ge-0/0/5 unit 0 family bridge interface-mode access
set forwarding-options analyzer employee-monitor input ingress bridge-domain remote-analyzer
set forwarding-options analyzer employee-monitor output interface ge-0/0/5.0
```

Step-by-Step Procedure

To configure basic remote mirroring using MX Series routers:

1. On the source switching device, do the following:

   - Configure the VLAN ID for the `remote-analyzer` bridge domain.

     ```
     [edit]
     user@device# set bridge-domains remote-analyzer vlan-id 999
     ```

   - Configure the interface on the network port connected to the destination switching device for access mode and associate it with the `remote-analyzer` bridge domain.

     ```
     [edit]
     user@device# set interfaces ge-0/0/10 unit 0 family bridge interface-mode access
     user@device# set interfaces ge-0/0/10 unit 0 family bridge vlan members 999
     ```

   - Configure the statistical analyzer `employee-monitor`.

     ```
     [edit forwarding-options]
     user@device# set analyzer employee-monitor input ingress interface ge-0/0/0.0
     user@device# set analyzer employee-monitor input ingress interface ge-0/0/1.0
     user@device# set analyzer employee-monitor input egress interface ge-0/0/0.0
     user@device# set analyzer employee-monitor input egress interface ge-0/0/1.0
     user@device# set analyzer employee-monitor output bridge-domain remote-analyzer
     user@device# set analyzer employee-monitor input rate 2
     user@device# set analyzer employee-monitor input maximum-packet-length 128
     ```

   - Bind the statistical analyzer to the FPC that contains the input interface.

     ```
     [edit]
     user@device# set chassis fpc 0 port-mirror-instance employee-monitor
     ```

2. On the destination switching device, do the following:

- Configure the VLAN ID for the `remote-analyzer` bridge domain.

```
[edit bridge-domains]
user@device# set remote-analyzer vlan-id 999
```

- Configure the interface on the destination switching device for access mode and associate it with the `remote-analyzer` bridge domain.

```
[edit interfaces]
user@device# set ge-0/0/10 unit 0 family bridge interface-mode access
user@device# set ge-0/0/10 unit 0 family bridge vlan members 999
```

- Configure the interface connected to the destination switching device for access mode.

```
[edit interfaces]
user@device# set ge-0/0/5 unit 0 family bridge interface-mode access
```

- Configure the `employee-monitor` analyzer.

```
[edit forwarding-options]
user@device# set analyzer employee-monitor input ingress bridge-domain remote-analyzer
user@device# set analyzer employee-monitor output interface ge-0/0/5.0
```

- Specify mirroring parameters such as rate and the maximum packet length for the `employee-monitor` analyzer.

```
[edit]
user@device# set forwarding-options analyzer employee-monitor input rate 2
user@device# set forwarding-options analyzer employee-monitor input maximum-packet-length
128
```

- Bind the `employee-monitor` analyzer to the FPC containing the input ports.

```
[edit]
user@device# set chassis fpc 0 port-mirror-instance employee-monitor
```

Results

Check the results of the configuration on the source switching device:

```
[edit]
user@device# show
bridge-domains {
    remote-analyzer {
        vlan-id 999;
    }
}
forwarding-options {
    analyzer {
        employee-monitor {
            input {
                ingress {
                    interface ge-0/0/0.0;
                    interface ge-0/0/1.0;
                }
                egress {
                    interface ge-0/0/0.0;
                    interface ge-0/0/1.0;
                }
                maximum-packet-length 128;
                rate 2;
            }
            output {
                bridge-domain {
                    remote-analyzer;
                }
            }
        }
    }
}
interfaces {
    ge-0/0/0 {
        unit 0 {
            family bridge {
                interface-mode access;
                vlan-id 99;
            }
        }
```

```
    }
    ge-0/0/1 {
        unit 0 {
            family bridge {
                interface-mode access;
                vlan-id 98;
            }
        }
    }
    ge-0/0/10 {
        unit 0 {
            family bridge {
                interface-mode access;
                vlan-id 999;
            }
        }
    }
}
```

Check the results of the configuration on the destination switching device.

```
[edit]
user@device# show
bridge-domains {
    remote-analyzer {
        vlan-id 999;
    }
}
forwarding-options {
    analyzer {
        employee-monitor {
            input {
                ingress {
                    interface ge-0/0/0.0;
                    interface ge-0/0/1.0;
                    bridge-domain remote-analyzer;
                }
            }
            output {
                interface ge-0/0/5.0;
            }
        }
```

```
    }
}
interfaces {
    ge-0/0/5 {
        unit 0 {
            family bridge {
                interface-mode access;
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1066**

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer named `employee-monitor` has been created on the device with the appropriate input interfaces and the appropriate output interface.

Action

To verify that the analyzer is configured as expected while monitoring all employee traffic on the source switching device, run the `show forwarding-options analyzer` command on the source switching device. The following output is displayed for this configuration example.

```
user@device> show forwarding-options analyzer

Analyzer name                    : employee-monitor
  Mirror rate                        : 2
  Maximum packet length            : 128
  State                              : up
  Ingress monitored interfaces     : ge-0/0/0.0
```

```
Ingress monitored interfaces     : ge-0/0/1.0
Egress monitored interfaces      : ge-0/0/0.0
Egress monitored interfaces      : ge-0/0/1.0
Output VLAN                      : default-switch/remote-analyzer
```

Meaning

This output shows that the `employee-monitor` instance has a ratio of 2, the maximum size of the original packet that were mirrored is 128, the state of the configuration is `up`, which indicates proper state and that the analyzer is programmed, and the analyzer is mirroring the traffic entering ge-0/0/0.0 and ge-0/0/1.0, and is sending the mirrored traffic to the VLAN called remote-analyzer.

If the state of the output interface is `down` or if the output interface is not configured, the value of `State` will be down and the analyzer will not be able to monitor traffic.

Example: Configuring Mirroring to Multiple Interfaces for Remote Monitoring of Employee Resource Use on EX9200 Switches

IN THIS SECTION

- Requirements | **1068**
- Overview and Topology | **1068**
- Mirroring All Employee Traffic to Multiple VLAN Member Interfaces for Remote Analysis | **1070**
- Verification | **1077**

EX9200 switches allow you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering or exiting a VLAN on

You can analyze the mirrored traffic using a protocol analyzer application running on a remote monitoring station if you are sending mirrored traffic to an analyzer VLAN.

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured mirroring analyzers when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by:
  - Using statistical sampling.
  - Setting ratios to select statistical samples.
  - Using firewall filters.

This example describes how to configure remote mirroring to multiple interfaces on an analyzer VLAN:

Requirements

This example uses the following hardware and software components:

- Three EX9200 switches
- Junos OS Release 13.2 or later for EX Series switches

Before you configure remote mirroring, be sure that:

- The interfaces that the analyzer will use as input interfaces have been configured on the switch.

Overview and Topology

IN THIS SECTION

Topology | **1070**

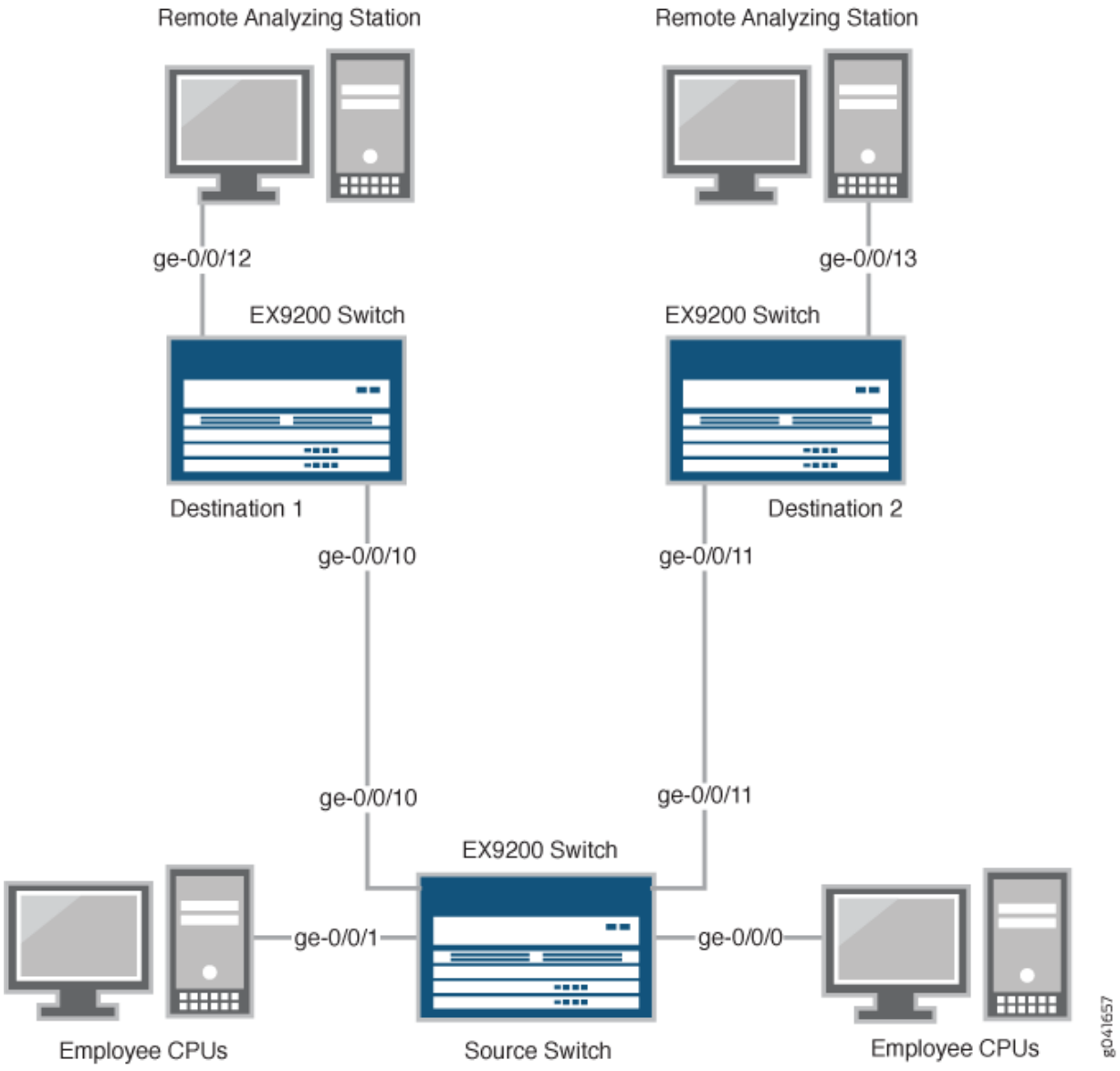
This example describes how to mirror traffic entering ports on the switch to the remote analyzer VLAN so that you can perform analysis from a remote monitoring station. The remote-analyzer VLAN in this example contains multiple member interfaces. Therefore, the same traffic is mirrored to all member interfaces of the remote-analyzer VLAN so that mirrored packets can be sent to different remote monitoring stations. You can install applications, such as sniffers and intrusion detection systems, on remote monitoring stations to analyze these mirrored packets and to obtain useful statistical data. For instance, if there are two remote monitoring stations, you can install a sniffer on one remote monitoring

station and an intrusion detection system on the other station. You can use a firewall filter analyzer configuration to forward a specific type of traffic to a remote monitoring station.

This example describes how to configure an analyzer to mirror traffic to multiple interfaces in the next-hop group so that traffic is sent to different monitoring stations for analysis.

Figure 36 on page 1069 shows the network topology for this example.

Figure 36: Remote Mirroring Example Network Topology Using Multiple VLAN Member Interfaces in the Next-Hop Group



Topology

In this example:

- Interfaces ge-0/0/0 and ge-0/0/1 are Layer 2 interfaces (both interfaces on the source switch) that serve as connections for employee computers.
- Interfaces ge-0/0/10 and ge-0/0/11 are Layer 2 interfaces that are connected to different destination switches.
- Interface ge-0/0/12 is a Layer 2 interface that connects the Destination 1 switch to the remote monitoring station.
- Interface ge-0/0/13 is a Layer 2 interface that connects the Destination 2 switch to the remote monitoring station.
- VLAN `remote-analyzer` is configured on all switches in the topology to carry the mirrored traffic.

Mirroring All Employee Traffic to Multiple VLAN Member Interfaces for Remote Analysis

IN THIS SECTION

- Procedure | **1070**

To configure mirroring to multiple VLAN member interfaces for remote traffic analysis for all incoming and outgoing employee traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure mirroring for remote traffic analysis for incoming and outgoing employee traffic, copy the following commands and paste them into the switch terminal window:

- In the source switch terminal window, copy and paste the following commands:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode access
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

```
 set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode access
 set interfaces ge-0/0/11 unit 0 family ethernet-switching vlan members 999
 set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
 set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
 set forwarding-options analyzer employee-monitor input egress interface ge-0/0/0.0
 set forwarding-options analyzer employee-monitor input egress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output next-hop-group remote-analyzer-nhg
set forwarding-options next-hop-group remote-analyzer-nhg interface ge-0/0/10.0
set forwarding-options next-hop-group remote-analyzer-nhg interface ge-0/0/11.0
set forwarding-options next-hop-group remote-analyzer-nhg group-type layer-2
```

- In the Destination 1 switch terminal window, copy and paste the following commands:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode acess
 set interfaces ge-0/0/12 unit 0 family ethernet-switching interface-mode access
set forwarding-options analyzer employee-monitor input ingress vlan remote-analyzer
set forwarding-options analyzer employee-monitor loss-priority high output interface
ge-0/0/12.0
```

- In the Destination 2 switch terminal window, copy and paste the following commands:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode access
 set interfaces ge-0/0/13 unit 0 family ethernet-switching interface-mode access
set forwarding-options analyzer employee-monitor input ingress vlan remote-analyzer
set forwarding-options analyzer employee-monitor loss-priority high output interface
ge-0/0/13.0
```

Step-by-Step Procedure

To configure basic remote mirroring to two VLAN member interfaces:

1. On the source switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interfaces on the network port connected to destination switches for access mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
user@switch# set ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/11 unit 0 family ethernet-switching vlan members 999
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
user@switch# set analyzer employee-monitor input egress interface ge-0/0/0.0
user@switch# set analyzer employee-monitor input egress interface ge-0/0/1.0
user@switch# set analyzer employee-monitor output next-hop-group remote-analyzer-nhg
```

  In this analyzer configuration, traffic that enters and exits interfaces ge-0/0/0.0 and ge-0/0/1.0 are sent to the output destination defined by the next-hop group named `remote-analyzer-nhg`.

- Configure the `remote-analyzer-nhb` next-hop group:

```
[edit forwarding-options]
user@switch# set next-hop-group remote-analyzer-nhg interface ge-0/0/10.0
user@switch# set next-hop-group remote-analyzer-nhg interface ge-0/0/11.0
user@switch# set next-hop-group remote-analyzer-nhg group-type layer-2
```

2. On the Destination 1 switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the ge-0/0/10 interface on the Destination 1 switch for access mode:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode access
```

- Configure the interface connected to the remote monitoring station for access mode:

```
[edit interfaces]
user@switch# set ge-0/0/12 unit 0 family ethernet-switching interface-mode access
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress vlan remote-analyzer
user@switch# set analyzer employee-monitor loss-priority high output interface ge-0/0/12.0
```

3. On the Destination 2 switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the ge-0/0/11 interface on the Destination 2 switch for access mode:

```
[edit interfaces]
user@switch# set ge-0/0/11 unit 0 family ethernet-switching interface-mode access
```

- Configure the interface connected to the remote monitoring station for access mode:

```
[edit interfaces]
user@switch# set ge-0/0/13 unit 0 family ethernet-switching interface-mode access
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress vlan remote-analyzer
user@switch# set analyzer employee-monitor loss-priority high output interface ge-0/0/13.0
```

Results

Check the results of the configuration on the source switch:

```
[edit]
user@switch# show
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
            egress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
        }
        output {
            next-hop-group {
                remote-analyzer-nhg;
            }
        }
    }
}
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
```

```
            ge-0/0/10.0
            ge-0/0/11.0
        }
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
    ge-0/0/11 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
}
```

Check the results of the configuration on the Destination 1 switch:

```
[edit]
user@switch# show
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            ethernet-switching {
                interface-mode acess;
            }
        }
    }
    ge-0/0/12 {
        unit 0 {
            family ethernet-switching {
```

```
            interface-mode access;
            }
        }
    }
}
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
            vlan remote-analyzer;
            }
        }
        loss-priority high;
        output {
            interface {
                ge-0/0/12.0;
            }
        }
    }
}
```

Check the results of the configuration on the Destination 2 switch:

```
[edit]
user@switch# show
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/11.0
        }
    }
}
interfaces {
    ge-0/0/11 {
        unit 0 {
        family ethernet-switching {
            interface-mode access;
        }
        }
    }
    ge-0/0/13 {
```

```
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
}
forwarding-options {
    employee-monitor {
        input {
            ingress {
                vlan remote-analyzer;
            }
        }
        loss-priority high;
        output {
            interface {
                ge-0/0/13.0;
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | 1077

To confirm that the configuration is working properly, perform these tasks:

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer named `employee-monitor` has been created on the switch with the appropriate input interfaces and appropriate output interface.

Action

You can verify the analyzer is configured as expected by using the `show forwarding-options analyzer` command.

To verify that the analyzer is configured as expected while monitoring all employee traffic on the source switch, run the `show forwarding-options analyzer` command on the source switch. The following output is displayed for this example configuration on the source switch:

```
user@switch> show forwarding-options analyzer
  Analyzer name                    : employee-monitor
  Mirror rate                         : 1
  Maximum packet length            : 0
  State                               : up
  Ingress monitored interfaces     : ge-0/0/0.0
  Ingress monitored interfaces     : ge-0/0/1.0
  Egress monitored interfaces     : ge-0/0/0.0
  Egress monitored interfaces     : ge-0/0/1.0
  Output nhg                       : remote-analyzer-nhg
user@switch> show forwarding-options next-hop-group
Next-hop-group: remote-analyzer-nhg
  Type:  layer-2
  State: up
  Members Interfaces:
    ge-0/0/10.0
    ge-0/0/11.0
```

Meaning

This output shows that the `employee-monitor` analyzer has a ratio of 1 (mirroring every packet, which is the default behavior), the state of the configuration is `up`, which indicates proper state and that the analyzer is programmed, mirrors traffic entering or exiting interfaces ge-0/0/0 and ge-0/0/1, and sends mirrored traffic to multiple interfaces ge-0/0/10.0 and ge-0/0/11.0 through the next-hop-group `remote-analyzer-nhg`. If the state of the output interface is `down` or if the output interface is not configured, the value of state will be down and the analyzer will not be able to mirror traffic.

Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use Through a Transit Switch on EX9200 Switches

IN THIS SECTION

- Requirements | 1080
- Overview and Topology | 1080
- Mirroring All Employee Traffic for Remote Analysis Through a Transit Switch | 1082
- Verification | 1088

EX9200 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering or exiting a VLAN

You can analyze the mirrored traffic using a protocol analyzer application running on a remote monitoring station if you are sending mirrored traffic to an analyzer VLAN.

This topic includes an example that describes how to mirror traffic entering ports on the switch to the remote-analyzer VLAN through a transit switch, so that you can perform analysis from a remote monitoring station.

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured mirroring sessions when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by:
  - Using statistical sampling.
  - Setting ratios to select statistical samples.

- Using firewall filters.

This example describes how to configure remote mirroring through a transit switch:

Requirements

This example uses the following hardware and software components:

- An EX9200 switch connected to another EX9200 switch through a third EX9200 switch
- Junos OS Release 13.2 or later for EX Series switches

Before you configure remote mirroring, be sure that:

- The interfaces that the analyzer will use as input interfaces have been configured on the switch.

Overview and Topology

IN THIS SECTION

- Topology | **1081**

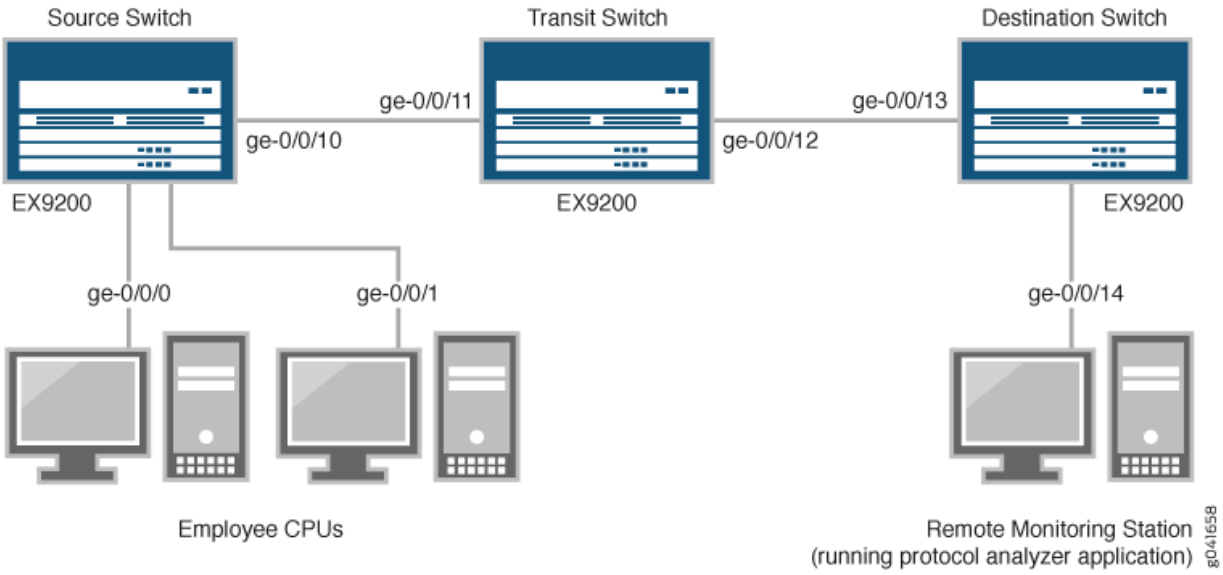
This example describes how to mirror traffic entering ports on the switch to the `remote-analyzer` VLAN through a transit switch so that you can perform analysis on all traffic from employee computers.

In this configuration, an analyzer session is required on the destination switch to mirror incoming traffic from the analyzer VLAN to the egress interface to which the remote monitoring station is connected.

Figure 37 on page 1081 shows the network topology for this example.

Topology

Figure 37: Network Monitoring for Remote Mirroring Through a Transit Switch



In this example:

1. Interface ge-0/0/0 is a Layer 2 interface, and interface ge-0/0/1 is a Layer 3 interface (both interfaces on the source switch) that serve as connections for employee computers.
2. Interface ge-0/0/10 is a Layer 2 interface that connects to the transit switch.
3. Interface ge-0/0/11 is a Layer 2 interface on the transit switch.
4. Interface ge-0/0/12 is a Layer 2 interface on the transit switch and connects to the destination switch.
5. Interface ge-0/0/13 is a Layer 2 interface on the destination switch.
6. Interface ge-0/0/14 is a Layer 2 interface on the destination switch and connects to the remote monitoring station.
7. VLAN `remote-analyzer` is configured on all switches in the topology to carry the mirrored traffic.

Mirroring All Employee Traffic for Remote Analysis Through a Transit Switch

IN THIS SECTION

- Procedure | 1082

To configure mirroring for remote traffic analysis through a transit switch, for all incoming and outgoing employee traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure mirroring for remote traffic analysis through a transit switch, for incoming and outgoing employee traffic, copy the following commands and paste them into the switch terminal window:

- Copy and paste the following commands in the source switch (monitored switch) terminal window:

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output vlan remote-analyzer
```

- Copy and paste the following commands in the transit switch window:

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode access
set vlans remote-analyzer interface ge-0/0/11
set interfaces ge-0/0/12 unit 0 family ethernet-switching interface-mode access
set vlans remote-analyzer interface ge-0/0/12
```

- Copy and paste the following commands in the destination switch window:

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/13 unit 0 family ethernet-switching interface-mode access
set vlans remote-analyzer interface ge-0/0/13 ingress
set interfaces ge-0/0/14 unit 0 family ethernet-switching interface-mode access
set forwarding-options analyzer employee-monitor input ingress vlan remote-analyzer
set forwarding-options analyzer employee-monitor output interface ge-0/0/14.0
```

Step-by-Step Procedure

To configure remote mirroring through a transit switch:

1. On the source switch:

   - Configure the VLAN ID for the `remote-analyzer` VLAN:

     ```
     [edit vlans]
     user@switch# set remote-analyzer vlan-id 999
     ```

   - Configure the interfaces on the network port connected to transit switch for access mode and associate it with the `remote-analyzer` VLAN:

     ```
     [edit interfaces]
     user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode access
     user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
     ```

   - Configure the `employee-monitor` analyzer:

     ```
     [edit forwarding-options]
     user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
     user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
     user@switch# set analyzer employee-monitor input egress interface ge-0/0/0.0
     user@switch# set analyzer employee-monitor input egress interface ge-0/0/1.0
     user@switch# set analyzer employee-monitor output vlan remote-analyzer
     ```

2. On the transit switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the ge-0/0/11 interface for access mode, associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/11 unit 0 family ethernet-switching interface-mode access
```

- Configure the ge-0/0/12 interface for access mode, associate it with the `remote-analyzer` VLAN, and set the interface for egress traffic only:

```
[edit interfaces]
user@switch# set ge-0/0/12 unit 0 family ethernet-switching interface-mode access
user@switch# set vlans remote-analyzer interface ge-0/0/12
```

3. On the destination switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the ge-0/0/13 interface for access mode, associate it with the `remote-analyzer` VLAN, and set the interface for ingress traffic only:

```
[edit interfaces]
user@switch# set ge-0/0/13 unit 0 family ethernet-switching interface-mode access
user@switch# set vlans remote-analyzer interface ge-0/0/13 ingress
```

- Configure the interface connected to the remote monitoring station for access mode:

```
[edit interfaces]
user@switch# set ge-0/0/14 unit 0 family ethernet-switching interface-mode access
```

- Configure the `remote-analyzer` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress vlan remote-analyzer
user@switch# set analyzer employee-monitor output interface ge-0/0/14.0
```

Results

Check the results of the configuration on the source switch:

```
[edit]
user@switch> show
 forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
            egress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
        }
        output {
            vlan {
                remote-analyzer;
            }
        }
    }
}
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
```

```
                vlan {
                    member 999;
                }
            }
        }
    }
}
```

Check the results of the configuration on the transit switch:

```
[edit]
user@switch> show
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/11.0 {
            }
            ge-0/0/12.0 {
            }
        }
    }
}
interfaces {
    ge-0/0/11 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
    ge-0/0/12 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
}
```

Check the results of the configuration on the destination switch:

```
[edit]
user@switch> show
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/13.0 {
                ingress;
            }
        }
    }
}
interfaces {
    ge-0/0/13 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
    ge-0/0/14 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
            }
        }
    }
}
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                vlan remote-analyzer;
            }
        }
        output {
            interface {
                ge-0/0/14.0;
            }
        }
```

```
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1088**

To confirm that the configuration is working properly, perform these tasks:

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer named `employee-monitor` has been created on the switch with the appropriate input interfaces and the appropriate output interface.

Action

You can verify the analyzer is configured as expected by using the `show forwarding-options analyzer` command.

To verify that the analyzer is configured as expected while monitoring all employee traffic on the source switch, run the `show forwarding-options analyzer` command on the source switch. The following output is displayed for this example configuration:

```
user@switch> show forwarding-options analyzer
Analyzer name                    : employee-monitor
  Mirror rate                      : 1
  Maximum packet length            : 0
  State                            : up
  Ingress monitored interfaces     : ge-0/0/0.0
  Ingress monitored interfaces     : ge-0/0/1.0
  Egress monitored interfaces      : ge-0/0/0.0
  Egress monitored interfaces      : ge-0/0/1.0
  Output vlan                      : default-switch/remote-analyzer
```

Meaning

This output shows that the `employee-monitor` analyzer has a mirroring ratio of 1 (mirroring every packet, the default), the state of the configuration is `up`, which indicates proper state and that the analyzer is programmed, is mirroring the traffic entering ge-0/0/0 and ge-0/0/1, and is sending the mirrored traffic to the analyzer called `remote-analyzer`. If the state of the output interface is `down` or if the output interface is not configured, the value of state will be down and the analyzer will not be able to mirror traffic.

Example: Configuring Mirroring for Local Monitoring of Employee Resource Use on EX4300 Switches

IN THIS SECTION

- Requirements | **1090**
- Overview and Topology | **1090**
- Mirroring All Employee Traffic for Local Analysis | **1091**
- Mirroring Employee-to-Web Traffic for Local Analysis | **1093**
- Verification | **1097**

NOTE: This example uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see Example: Configuring Port Mirroring for Local Monitoring of Employee Resource Use on EX Series Switches. For ELS details, see *Getting Started with Enhanced Layer 2 Software*.

EX4300 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering a VLAN

You can analyze the mirrored traffic by using a protocol analyzer installed on a system connected to the local destination interface or a remote monitoring station if you are sending mirrored traffic to an analyzer VLAN.

This example describes how to configure local mirroring on an EX4300 switch. This example describes how to configure the switch to mirror traffic entering interfaces connected to employee computers to an analyzer output interface on the same switch.

Requirements

This example uses the following hardware and software components:

- One EX4300 switch
- Junos OS Release 13.2X50-D10 or later for EX Series switches

Overview and Topology

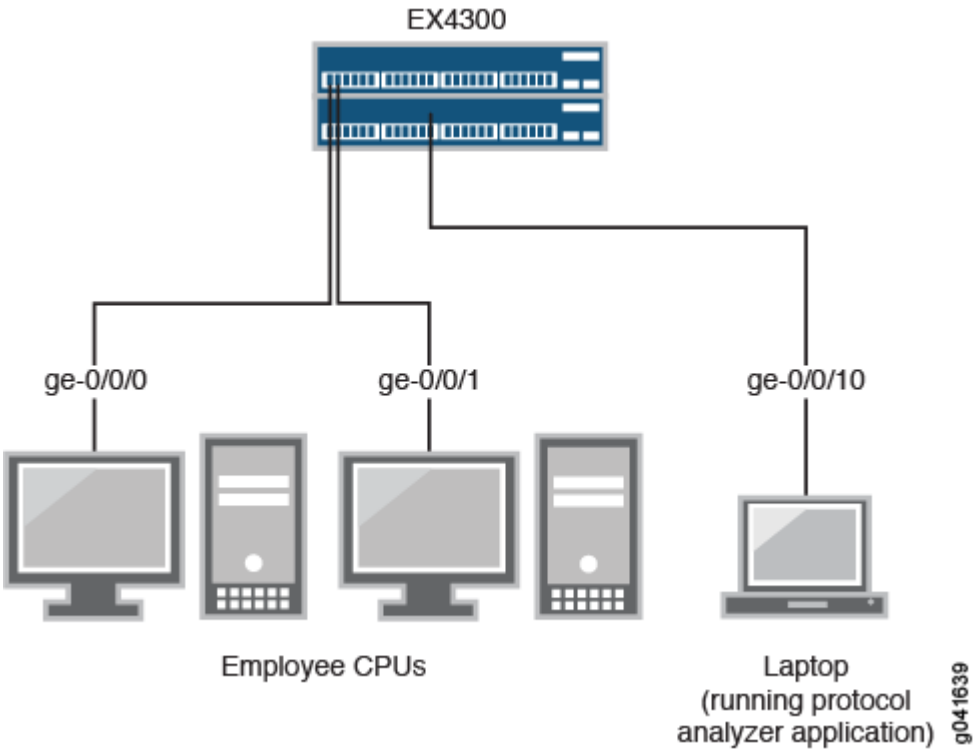
This topic includes two examples that describe how to mirror traffic entering ports on the switch to a destination interface on the same switch (local mirroring). The first example shows how to mirror all traffic entering the ports connected to employee computers. The second example shows the same scenario, but includes a filter to mirror only the employee traffic going to the Web.

The interfaces ge-0/0/0 and ge-0/0/1 serve as connections for employee computers. The interface ge0/0/10 is reserved for analysis of mirrored traffic. Connect a PC running a protocol analyzer application to the analyzer output interface to analyze the mirrored traffic.

NOTE: Multiple ports mirrored to one interface can cause buffer overflow and dropped packets.

Both examples use the network topology shown in Figure 38 on page 1091.

Figure 38: Network Topology for Local Mirroring Example



Mirroring All Employee Traffic for Local Analysis

IN THIS SECTION

- Procedure | 1092

To configure mirroring for all employee traffic for local analysis, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure local mirroring for ingress traffic to the two ports connected to employee computers, copy the following commands and paste them into the switch terminal window:

```
[edit]
set interfaces ge-0/0/0 unit 0 family ethernet-switching
set interfaces ge-0/0/1 unit 0 family inet address 192.0.2.1/24
set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members analyzer_vlan
set vlans analyzer-vlan vlan-id 1000
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output interface ge-0/0/10.0
```

Step-by-Step Procedure

To configure an analyzer called `employee-monitor` and specify the input (source) interfaces and the analyzer output interface:

1. Configure each interface connected to employee computers as an input interface for the analyzer `employee-monitor`:

   ```
   [edit forwarding-options]
   user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
   user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
   ```

2. Configure the output interface of the analyzer as part of a VLAN:

   ```
   [edit interfaces]
   user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members analyzer_vlan
   ```

   ```
   [edit vlans]
   user@switch# set analyzer-vlan vlan-id 1000
   ```

3. Configure the output analyzer interface for the analyzer `employee-monitor`. This will be the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output interface ge-0/0/10.0
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
            interface ge-0/0/1.0;}
        }
        output {
            interface {
                ge-0/0/10.0;
            }
        }
    }
}
```

Mirroring Employee-to-Web Traffic for Local Analysis

IN THIS SECTION

- Procedure | **1094**

To configure mirroring for employee to Web traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure local mirroring of traffic from the two ports connected to employee computers, filtering so that only traffic to the external Web is mirrored, copy the following commands and paste them into the switch terminal window:

```
[edit]
set forwarding-options port-mirroring instance employee-web-monitor output interface ge-0/0/10.0
 set firewall family ethernet-switching filter watch-employee term employee-to-corp from
destination-address 192.0.2.16/24
 set firewall family ethernet-switching filter watch-employee term employee-to-corp from source-
address 192.0.2.16/24
 set firewall family ethernet-switching filter watch-employee term employee-to-corp then accept
set firewall family ethernet-switching filter watch-employee term employee-to-web from
destination-port 80
set firewall family ethernet-switching filter watch-employee term employee-to-web then port-
mirroring-instance employee-web-monitor
set interfaces ge-0/0/0 unit 0 family ethernet-switching filter input watch-employee
 set interfaces ge-0/0/1  unit 0 family ethernet-switching filter input watch-employee
```

Step-by-Step Procedure

To configure local mirroring of employee to Web traffic from the two ports connected to employee computers:

1. Configure the local analyzer interface:

   ```
   [edit interfaces]
   user@switch# set ge-0/0/10 unit 0 family ethernet-switching
   ```

2. Configure the `employee-web-monitor` output instance (the input to the instance comes from the action of the filter):

   ```
   [edit forwarding-options port-mirroring]
   user@switch# set instance employee-web-monitor output interface ge-0/0/10.0
   ```

3. Configure a firewall filter called `watch-employee` to send mirrored copies of employee requests to the Web to the `employee-web-monitor` instance. Accept all traffic to and from the corporate subnet (destination or source address of 192.0.2.16/24). Send mirrored copies of all packets destined for the Internet (destination port 80) to the `employee-web-monitor` instance.

```
[edit firewall family ethernet-switching]
user@switch# set filter watch-employee term employee-to-corp from destination-address
192.0.2.16/24
user@switch# set filter watch-employee term employee-to-corp from source-address 192.0.2.16/24
user@switch# set filter watch-employee term employee-to-corp then accept
ser@switch# set filter watch-employee term employee-to-web from destination-port 80
user@switch# set filter watch-employee term employee-to-web then port-mirroring-instance
employee-web-monitor
```

4. Apply the `watch-employee` filter to the appropriate ports:

```
[edit interfaces]
user@switch# set ge-0/0/0 unit 0 family ethernet-switching filter input watch-employee
user@switch# set ge-0/0/1 unit 0 family ethernet-switching filter input watch-employee
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show
 forwarding-options {
    port-mirroring {
        instance     {
            employee-web-monitor {
                family ethernet-switching {
                    output {
                        interface ge-0/0/10.0;
                        }
                    }
                }
            }
    }
}
```

```
...
firewall family ethernet-switching {
    filter watch-employee {
        term employee-to-corp {
            from {
                destination-address 192.0.2.16/24;
                source-address 192.0.2.16/24;
            }
            then accept {
        }
        term employee-to-web {
            from {
                destination-port 80;
            }
            then port-mirroring-instance employee-web-monitor;
        }
    }
}
...
interfaces {
    ge-0/0/0 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
                vlan members [employee-vlan, voice-vlan];
                filter {
                    input watch-employee;
                }
            }
        }
    }
    ge-0/0/1 {
        family ethernet-switching {
            filter {
                input watch-employee;
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | 1097
- Verifying That The Port-Mirroring Instance Is Configured Properly | 1098

To confirm that the configuration is correct, perform these tasks:

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer `employee-monitor` or `employee-web-monitor` has been created on the switch with the appropriate input interfaces, and appropriate output interface.

Action

You can use the `show forwarding-options analyzer` command to verify that the analyzer is configured properly.

```
user@switch> show forwarding-options analyzer
  Analyzer name                          : employee-monitor
  Mirror rate                           : 1
  Maximum packet length              : 0
  State                                  : up
  Ingress monitored interfaces     : ge-0/0/0.0
  Ingress monitored interfaces     : ge-0/0/1.0
  Output interface                      : ge-0/0/10.0
```

Meaning

This output shows that the analyzer `employee-monitor` has a ratio of 1 (mirroring every packet, the default setting), the maximum size of the original packet that was mirrored (`0` indicates the entire packet), the state of the configuration (is up indicates that the analyzer is mirroring the traffic entering the ge-0/0/0,

and ge-0/0/1 interfaces, and sending the mirrored traffic to the ge-0/0/10 interface). If the state of the output interface is down or if the output interface is not configured, the value of state will be `down` and the analyzer will not be programmed for mirroring.

Verifying That The Port-Mirroring Instance Is Configured Properly

Purpose

Verify that the port-mirroring instance `employee-web-monitor` has been configured properly on the switch with the appropriate input interfaces.

Action

You can verify that the port-mirroring instance is configured properly by using the `show forwarding-options port-mirroring` command.

```
user@switch> show forwarding-options port-mirroring
Instance Name: employee-web-monitor
  Instance Id: 3
  Input parameters:
    Rate                  : 1
    Run-length            : 0
    Maximum-packet-length : 0
  Output parameters:
    Family              State     Destination          Next-hop
    ethernet-switching  up        ge-0/0/10.0
```

Meaning

This output shows that the `employee-web-monitor` instance has a ratio of 1 (mirroring every packet, the default), the maximum size of the original packet that was mirrored (`0` indicates an entire packet), the state of the configuration is up and port mirroring is programmed, and that mirrored traffic from the firewall filter action is sent out on interface ge-0/0/10.0. If the state of the output interface is down or if the interface is not configured, the value for state will be down and port mirroring will not be programmed for mirroring.

Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use on EX4300 Switches

IN THIS SECTION

- Requirements | **1100**
- Overview and Topology | **1100**
- Mirroring All Employee Traffic for Remote Analysis | **1101**
- Mirroring Employee-to-Web Traffic for Remote Analysis | **1106**
- Verification | **1112**

NOTE: This example uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see "Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use on EX4300 Switches" on page 1099. For ELS details see: *Getting Started with Enhanced Layer 2 Software*.

EX4300 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering a VLAN on EX4300 switches

You can analyze the mirrored traffic by using a protocol analyzer application running on a remote monitoring station if you are sending mirrored traffic to an analyzer VLAN.

This topic includes two related examples that describe how to mirror traffic entering ports on the switch to the `remote-analyzer` VLAN so that you can perform analysis from a remote monitoring station. The first example shows how to mirror all traffic entering the ports connected to employee computers. The second example shows the same scenario but includes a filter to mirror only the employee traffic going to the Web.

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured mirroring sessions when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by using firewall filters.

This example describes how to configure remote mirroring:

Requirements

This example uses the following hardware and software components:

- Junos OS Release 13.2X50-D10 or later for EX Series switches
- An EX4300 switch connected to another EX4300 switch

The diagram shows an EX4300 Virtual Chassis connected to an EX4300 destination switch.

Before you configure remote mirroring, be sure that:

- You have an understanding of mirroring concepts.
- The interfaces that the analyzer will use as input interfaces have been configured on the switch.

Overview and Topology

IN THIS SECTION

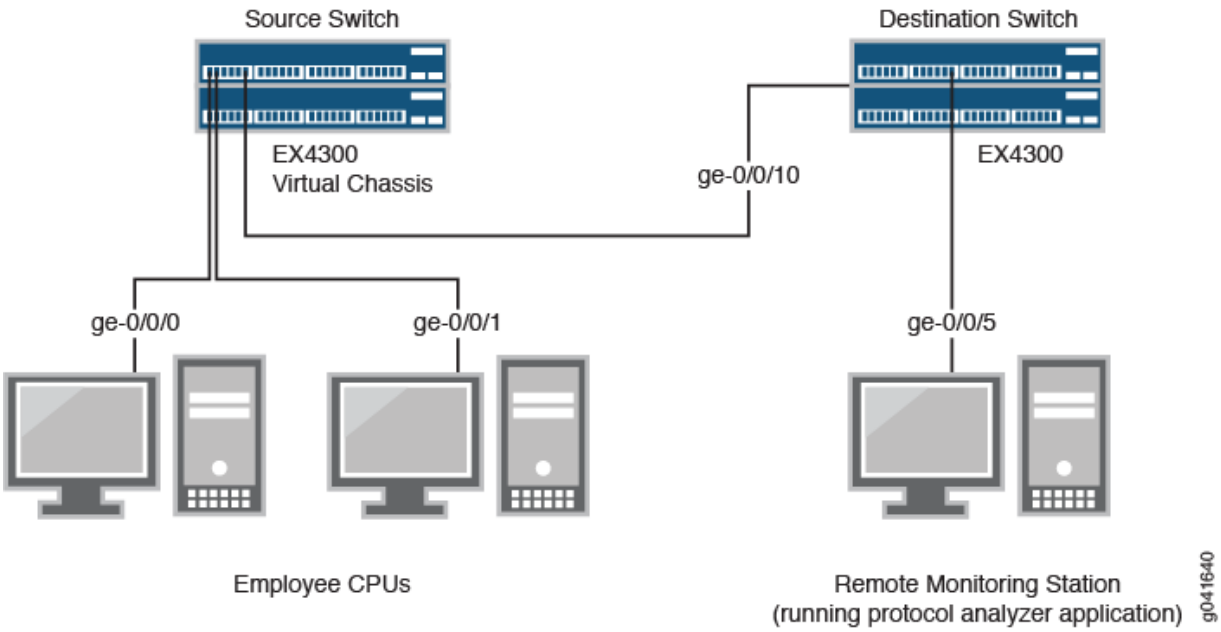
- Topology | **1101**

This topic includes two related examples that describe how to configure mirroring to the `remote-analyzer` VLAN so that analysis can be performed from a remote monitoring station. The first example shows how to configure a switch to mirror all traffic from employee computers. The second example shows the same scenario, but the setup includes a filter to mirror only the employee traffic going to the Web.

Figure 39 on page 1101 shows the network topology for both these example scenarios.

Topology

Figure 39: Remote Mirroring Network Topology Example



In this example:

1. Interface ge-0/0/0 is a Layer 2 interface, and interface ge-0/0/1 is a Layer 3 interface (both interfaces on the source switch) that serve as connections for employee computers.
2. Interface ge-0/0/10 is a Layer 2 interface that connects the source switch to the destination switch.
3. Interface ge-0/0/5 is a Layer 2 interface that connects the destination switch to the remote monitoring station.
4. VLAN `remote-analyzer` is configured on all switches in the topology to carry the mirrored traffic.

Mirroring All Employee Traffic for Remote Analysis

IN THIS SECTION

- Procedure | 1102

To configure an analyzer for remote traffic analysis for all incoming and outgoing employee traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure an analyzer for remote traffic analysis for incoming and outgoing employee traffic, copy the following commands and paste them into the switch terminal window:

- Copy and paste the following commands in the source switch terminal window:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
 set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
 set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
 set forwarding-options analyzer employee-monitor input egress interface ge-0/0/0.0
 set forwarding-options analyzer employee-monitor input egress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output vlan remote-analyzer
```

- Copy and paste the following commands in the destination switch terminal window:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
 set interfaces ge-0/0/5 unit 0 family ethernet-switching interface-mode trunk
set forwarding-options analyzer employee-monitor input ingress vlan remote-analyzer
set forwarding-options analyzer employee-monitor output interface ge-0/0/5.0
```

Step-by-Step Procedure

To configure basic remote port mirroring:

1. On the source switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interface on the network port connected to the destination switch for trunk mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
user@switch# set instance employee-monitor input egress interface ge-0/0/0.0
user@switch# set analyzer employee-monitor input egress interface ge-0/0/1.0
user@switch# set analyzer employee-monitor output vlan remote-analyzer
```

2. On the destination switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interface on the destination switch for trunk mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the interface connected to the destination switch for trunk mode:

```
[edit interfaces]
user@switch# set ge-0/0/5 unit 0 family ethernet-switching interface-mode trunk
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress vlan remote-analyzer
user@switch# set analyzer employee-monitor output interface ge-0/0/5.0
```

Results

Check the results of the configuration on the source switch:

```
[edit]
user@switch> show
 forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
            egress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
        }
        output {
            vlan {
                remote-analyzer;
            }
        }
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
```

```
                interface-mode trunk;
                vlan {
                    members 999;
                }
            }
        }
    }
}
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/10.0
            }
        }
    }
}
```

Check the results of the configuration on the destination switch:

```
[edit]
user@switch> show
interfaces {
    ge0/0/5 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
            }
        }
    }
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
                vlan {
                    members 999;
                }
            }
        }
    }
}
vlans {
```

```
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/10.0
            }
        }
    }
}
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                vlan remote-analyzer;
            }
        }
        output {
            interface {
                ge-0/0/5.0;
            }
        }
    }
}
```

Mirroring Employee-to-Web Traffic for Remote Analysis

IN THIS SECTION

- Procedure | **1106**

To configure port mirroring for remote traffic analysis of employee- to- Web traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure port mirroring to mirror employee traffic to the external Web, copy the following commands and paste them into the switch terminal window:

- Copy and paste the following commands in the source switch terminal window:

```
[edit]
user@switch# set forwarding-options port-mirroring instance employee-web-monitor output vlan
999
user@switch# set vlans remote-analyzer vlan-id 999
user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching port mode trunk
user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
user@switch# set firewall family ethernet-switching filter watch-employee term employee-to-
corp from destination-address 192.0.2.16/24
user@switch# set firewall family ethernet-switching filter watch-employee term employee-to-
corp from source-address 192.0.2.16/24
user@switch# set firewall family ethernet-switching filter watch-employee term employee-to-
corp then accept
user@switch# set firewall family ethernet-switching filter watch-employee term employee-to-
web from destination-port 80
user@switch# set firewall family ethernet-switching filter watch-employee term employee-to-
web then port-mirror-instance employee-web-monitor
user@switch# set interfaces ge-0/0/0 unit 0 family ethernet-switching filter input watch-
employee
user@switch# set interfaces ge-0/0/1 unit 0 family ethernet-switching filter input watch-
employee
```

- Copy and paste the following commands in the destination switch terminal window:

```
[edit]
user@switch# set vlans remote-analyzer vlan-id 999
user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
user@switch# set interfaces ge-0/0/5 unit 0 family ethernet-switching interface-mode trunk
user@switch# set forwarding-options analyzer employee-web-monitor input ingress vlan remote-
analyzer
user@switch# set forwarding-options analyzer employee-web-monitor output interface ge-0/0/5.0
```

Step-by-Step Procedure

To configure port mirroring of all traffic from the two ports connected to employee computers to the `remote-analyzer` VLAN for use from a remote monitoring station:

1. On the source switch:

- Configure the `employee-web-monitor` port mirroring instance:

```
[edit ]
user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching port mode trunk
user@switch# set forwarding-options port-mirroring instance employee-web-monitor output
vlan 999
```

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interface to associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the firewall filter called `watch-employee`:

```
[edit firewall family ethernet-switching]
user@switch# set filter watch-employee term employee-to-corp from destination-address
192.0.2.16/24
user@switch# set filter watch-employee term employee-to-corp from source-address
192.0.2.16/24
user@switch# set filter watch-employee term employee-to-corp then accept
user@switch# set filter watch-employee term employee-to-web from destination-port 80
user@switch# set filter watch-employee term employee-to-web then port-mirror-instance
employee-web-monitor
```

- Apply the firewall filter to the employee interfaces:

```
[edit interfaces]
user@switch# set ge-0/0/0 unit 0 family ethernet-switching filter input watch-employee
user@switch# set ge-0/0/1 unit 0 family ethernet-switching filter input watch-employee
```

2. On the destination switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interface on the destination switch for trunk mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the interface connected to the destination switch for trunk mode:

```
[edit interfaces]
user@switch# set ge-0/0/5 unit 0 family ethernet-switching interface-mode trunk
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options port-mirroring]
user@switch# set instance employee-web-monitor input ingress vlan remote-analyzer
user@switch# set instance employee-web-monitor output interface ge-0/0/5.0
```

Results

Check the results of the configuration on the source switch:

```
[edit]
user@switch> show
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
                vlan {
                    members remote-analyzer;
                }
```

```
            }
        }
    }
    ge-0/0/0 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
}
firewall {
     family ethernet-switching {
        filter watch-employee {
            term employee-to-corp {
                from {
                    source-address {
                        192.0.2.16/24;
                    }
                    destination-address {
                        192.0.2.16/24;
                    }
                }
                then accept;
            }
            term employee-to-web {
                from {
                    destination-port 80;
                }
                then port-mirror-instance employee-web-monitor;
            }
        }
```

```
    }
}
 forwarding-options {
    analyzer employee-web-monitor {
        output {
            vlan {
                999;
            }
        }
    }
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
```

Check the results of the configuration on the destination switch:

```
[edit]
user@switch> show
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
                vlan {
                    members remote-analyzer;
                }
            }
        }
    }
    ge-0/0/5 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
            }
        }
```

```
    }
}
 forwarding-options {
    port-mirroring {
        instance employee-web-monitor {
            input {
                ingress {
                    vlan remote-analyzer;
                }
            }
            output {
                interface {
                    ge-0/0/5.0;
                }
            }
        }
    }
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1112**

To confirm that the configuration is working properly, perform these tasks:

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer named `employee-monitor` or `employee-web-monitor` has been created on the switch with the appropriate input interfaces and appropriate output interface.

Action

You can verify the analyzer is configured as expected by using the `show forwarding-options analyzer` command. To view previously created analyzers that are disabled, go to the J-Web interface.

To verify that the analyzer is configured as expected while monitoring all employee traffic on the source switch, run the `show analyzer` command on the source switch. The following output is displayed for this configuration example:

```
user@switch> show forwarding-options analyzer
  Analyzer name                    : employee-monitor
  Mirror rate                           : 1
  Maximum packet length        : 0
  State                                 : up
  Ingress monitored interfaces     : ge-0/0/0.0
  Ingress monitored interfaces     : ge-0/0/1.0
  Egress monitored interfaces     : ge-0/0/0.0
  Egress monitored interfaces     : ge-0/0/1.0
  Output VLAN                        : default-switch/remote-analyzer
```

Meaning

This output shows that the `employee-monitor` instance has a ratio of 1 (mirroring every packet, the default), the maximum size of the original packet that was mirrored (0 indicates the entire packet), the state of the configuration is up (which indicates the proper state and that the analyzer is programmed, and is mirroring the traffic entering ge-0/0/0 and ge-0/0/1 and is sending the mirrored traffic to the VLAN called `remote-analyzer`). If the state of the output interface is down or if the output interface is not configured, the value of state will be down and the analyzer will not be programmed for mirroring.

Example: Configuring Mirroring for Remote Monitoring of Employee Resource Use Through a Transit Switch on EX4300 Switches

IN THIS SECTION

- Requirements | **1114**
- Overview and Topology | **1115**
- Mirroring All Employee Traffic for Remote Analysis Through a Transit Switch | **1116**
- Verification | **1122**

NOTE: This example uses Junos OS for EX Series switches with support for the Enhanced Layer 2 Software (ELS) configuration style.

EX4300 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering a VLAN on EX4300 switches

You can analyze the mirrored traffic by using a protocol analyzer application running on a remote monitoring station if you are sending mirrored traffic to an analyzer VLAN.

This topic includes an example that describes how to mirror traffic entering ports on the switch to the `remote-analyzer` VLAN through a transit switch, so that you can perform analysis from a remote monitoring station.

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured mirroring sessions when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by using firewall filters.

This example describes how to configure remote mirroring through a transit switch:

Requirements

This example uses the following hardware and software components:

- An EX4300 switch connected to another EX4300 switch through a third EX4300 switch
- Junos OS Release 13.2X50-D10 or later for EX Series switches

Before you configure remote mirroring, be sure that:

- You have an understanding of mirroring concepts.
- The interfaces that the analyzer will use as input interfaces have been configured on the switch.

Overview and Topology

IN THIS SECTION

- Topology | 1115

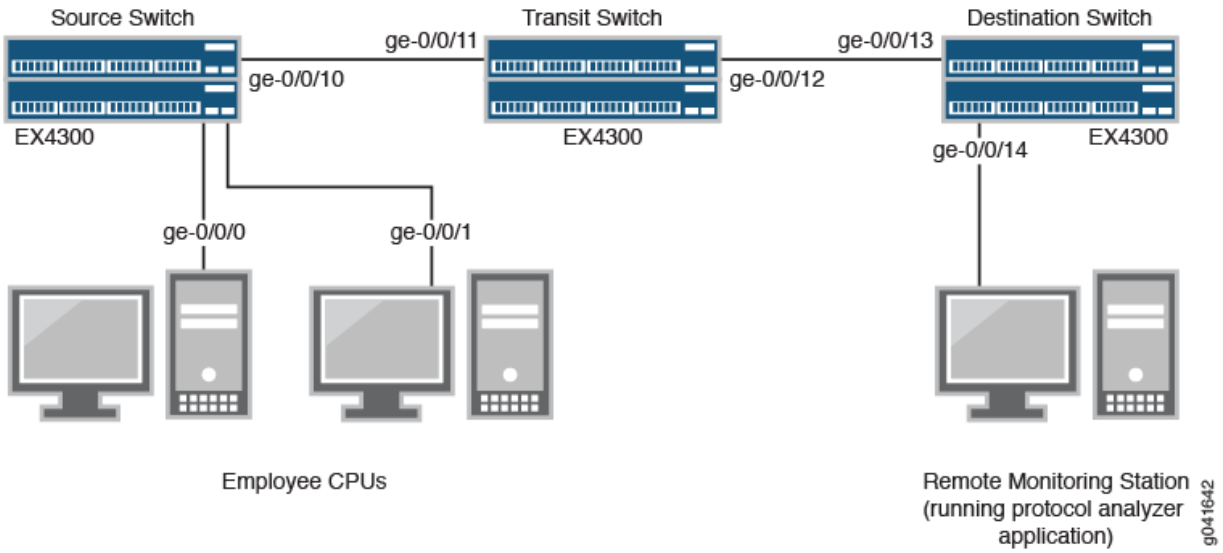
This example describes how to mirror traffic entering ports on the switch to the `remote-analyzer` VLAN through a transit switch so that you can perform analysis from a remote monitoring station. The example shows how to configure a switch to mirror all traffic from employee computers to a remote analyzer.

In this configuration, an analyzer session is required on the destination switch to mirror incoming traffic from the analyzer VLAN to the egress interface to which the remote monitoring station is connected. You must disable MAC learning on the transit switch for the `remote-analyzer` VLAN so that MAC learning is disabled for all member interfaces of the `remote-analyzer` VLAN on the transit switch.

Figure 40 on page 1115 shows the network topology for this example.

Topology

Figure 40: Remote Mirroring Through a Transit Switch Network–Sample Topology



In this example:

- Interface ge-0/0/0 is a Layer 2 interface, and interface ge-0/0/1 is a Layer 3 interface (both interfaces on the source switch) that serve as connections for employee computers.

- Interface ge-0/0/10 is a Layer 2 interface that connects to the transit switch.
- Interface ge-0/0/11 is a Layer 2 interface on the transit switch.
- Interface ge-0/0/12 is a Layer 2 interface on the transit switch and connects to the destination switch.
- Interface ge-0/0/13 is a Layer 2 interface on the destination switch .
- Interface ge-0/0/14 is a Layer 2 interface on the destination switch and connects to the remote monitoring station.
- VLAN `remote-analyzer` is configured on all switches in the topology to carry the mirrored traffic.

Mirroring All Employee Traffic for Remote Analysis Through a Transit Switch

IN THIS SECTION

- Procedure | **1116**

To configure mirroring for remote traffic analysis through a transit switch, for all incoming and outgoing employee traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure mirroring for remote traffic analysis through a transit switch, for incoming and outgoing employee traffic, copy the following commands and paste them into the switch terminal window:

- Copy and paste the following commands in the source switch (monitored switch) terminal window:

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/0.0
```

```
set forwarding-options analyzer employee-monitor input egress interface ge-0/0/1.0
set forwarding-options analyzer employee-monitor output vlan remote-analyzer
```

- Copy and paste the following commands in the transit switch window:

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
set vlans remote-analyzer interface ge-0/0/11
set interfaces ge-0/0/12 unit 0 family ethernet-switching interface-mode trunk
set vlans remote-analyzer interface ge-0/0/12
set vlans remote-analyzer no-mac-learning
```

- Copy and paste the following commands in the destination switch window:

```
[edit]
set vlans remote-analyzer vlan-id 999
set interfaces ge-0/0/13 unit 0 family ethernet-switching interface-mode trunk
set vlans remote-analyzer interface ge-0/0/13 ingress
set interfaces ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk
set forwarding-options analyzer employee-monitor input ingress vlan remote-analyzer
set forwarding-options analyzer employee-monitor output interface ge-0/0/14.0
```

Step-by-Step Procedure

To configure remote mirroring through a transit switch:

1. On the source switch:

   - Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interfaces on the network port connected to transit switch for trunk mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
user@switch# set analyzer employee-monitor input egress interface ge-0/0/0.0
user@switch# set analyzer employee-monitor input egress interface ge-0/0/1.0
user@switch# set analyzer employee-monitor output vlan remote-analyzer
```

2. On the transit switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the ge-0/0/11 interface for trunk mode, associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/11 unit 0 family ethernet-switching interface-mode trunk
```

- Configure the `ge-0/0/12` interface for trunk mode, associate it with the `remote-analyzer` VLAN, and set the interface for egress traffic only:

```
[edit interfaces]
user@switch# set ge-0/0/12 unit 0 family ethernet-switching interface-mode trunk
user@switch# set vlans remote-analyzer interface ge-0/0/12
```

- Configure the `no-mac-learning` option for the `remote-analyzer` VLAN to disable MAC learning on all interfaces that are members of the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set vlans remote-analyzer no-mac-learning
```

3. On the destination switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the ge-0/0/13 interface for trunk mode, associate it with the `remote-analyzer` VLAN, and set the interface for ingress traffic only:

```
[edit interfaces]
user@switch# set ge-0/0/13 unit 0 family ethernet-switching interface-mode trunk
user@switch# set vlans remote-analyzer interface ge-0/0/13 ingress
```

- Configure the interface connected to the remote monitoring station for trunk mode:

```
[edit interfaces]
user@switch# set ge-0/0/14 unit 0 family ethernet-switching interface-mode trunk
```

- Configure the `employee-monitor` analyzer:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor input ingress vlan remote-analyzer
user@switch# set analyzer employee-monitor output interface ge-0/0/14.0
```

Results

Check the results of the configuration on the source switch:

```
[edit]
user@switch> show
```

```
 forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
            egress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
        }
        output {
            vlan {
                remote-analyzer;
            }
        }
    }
}
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
                vlan {
                    member 999;
                }
            }
        }
    }
}
```

Check the results of the configuration on the transit switch:

```
[edit]
user@switch> show
vlans {
```

```
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/11.0 {
            }
            ge-0/0/12.0 {
            }
        }
        no-mac-learning;
    }
}
interfaces {
    ge-0/0/11 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
            }
        }
    }
    ge-0/0/12 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
            }
        }
    }
}
```

Check the results of the configuration on the destination switch:

```
[edit]
user@switch> show
vlans {
    remote-analyzer {
        vlan-id 999;
        interface {
            ge-0/0/13.0 {
                ingress;
            }
        }
    }
}
```

```
interfaces {
    ge-0/0/13 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
            }
        }
    }
    ge-0/0/14 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
            }
        }
    }
}
forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                vlan remote-analyzer;
            }
        }
        output {
            interface {
                ge-0/0/14.0;
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1123**

To confirm that the configuration is working properly, perform these tasks:

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer named `employee-monitor` has been created on the switch with the appropriate input interfaces and the appropriate output interface.

Action

You can verify whether the analyzer is configured as expected by using the `show analyzer` command. To view previously created analyzers that are disabled, go to the J-Web interface.

To verify that the analyzer is configured as expected while monitoring all employee traffic on the source switch, run the `show analyzer` command on the source switch. The following output is displayed for this example configuration:

```
user@switch> show forwarding-options analyzer
Analyzer name                    : employee-monitor
  Mirror rate                      : 1
  Maximum packet length            : 0
  State                            : up
  Ingress monitored interfaces     : ge-0/0/0.0
  Ingress monitored interfaces     : ge-0/0/1.0
  Egress monitored interfaces      : ge-0/0/0.0
  Egress monitored interfaces      : ge-0/0/1.0
  Output vlan                      : default-switch/remote-analyzer
```

Meaning

This output shows that the `employee-monitor` analyzer has a ratio of 1 (mirroring every packet, the default), is mirroring the traffic entering ge-0/0/0 and ge-0/0/1, and sending the mirrored traffic to the analyzer `remote-analyzer`.

Configuring Port Mirroring Instances

IN THIS SECTION

- Layer 2 Port Mirroring Global Instance | **1124**
- Configuring the Global Instance of Layer 2 Port Mirroring | **1124**
- Layer 2 Port Mirroring Named Instances | **1127**
- Defining a Named Instance of Layer 2 Port Mirroring | **1129**
- Disabling Layer 2 Port Mirroring Instances | **1133**
- Configuring Inline Port Mirroring | **1134**

Layer 2 Port Mirroring Global Instance

On an MX Series router and on an EX Series switch, you can configure a set of port-mirroring properties that implicitly apply to packets received on all ports in the router (or switch) chassis. This set of port-mirroring properties is the *global instance* of Layer 2 *port mirroring* for the router or switch.

Within the global instance configuration, you can specify a set of mirror destination properties for each packet address family supported by Layer 2 port mirroring.

For a general description of Layer 2 port-mirroring properties, see "Understanding Layer 2 Port Mirroring Properties" on page 1013. For a comparison of the types of Layer 2 port mirroring available on an MX Series router and on an EX Series switch, see Application of Layer 2 Port Mirroring Types.

Configuring the Global Instance of Layer 2 Port Mirroring

On an MX Series router and on an EX Series switch, you can configure a set of Layer 2 port-mirroring properties that implicitly apply to packets received on all ports in the router (or switch) chassis.

To configure the global instance of Layer 2 port mirroring on an MX Series router and on an EX Series switch:

1. Enable configuration of the Layer 2 port mirroring:

```
[edit]
user@host# edit forwarding-options port-mirroring
```

2. Enable configuration of the packet-selection properties:

```
[edit forwarding-options port-mirroring]
user@host# edit input
```

3. Specify global-level packet-selection properties.

   a. Specify the number of packets to select:

   ```
   [edit forwarding-options port-mirroring input]
   user@host# set rate number
   ```

   The valid range is 1 through 65535.

   b. Specify the number of packets to mirror from each selection:

   ```
   [edit forwarding-options port-mirroring input]
   user@host# set run-length number
   ```

   The valid range is 0 through 20. The default value is 0.

   c. Specify the length to which mirrored packets are to be truncated:

   ```
   [edit forwarding-options port-mirroring input]
   user@host# set maximum-packet-length number
   ```

   The valid range is 0 through 9216. The default value is 0, which means the mirrored packets are not truncated.

4. Specify the global-level Layer 2 address-type family from which traffic is to be selected for mirroring:

```
[edit forwarding-options port-mirroring input]
user@host# up
[edit forwarding-options port-mirroring]
user@host# edit family family
```

The value of the *family* option can be `ethernet-switching`, `ccc`or `vpls`.

NOTE: Under the `[edit forwarding-options port-mirroring]` hierarchy level, the protocol family statement `family ethernet-switching` is an alias for `family vpls`. The command-line interface (CLI) displays Layer 2 port-mirroring configurations as `family vpls`, even for Layer 2 port-mirroring configured as `family ethernet-switching`. Use `family ethernet-switching` when the physical interface is configured with `encapsulation ethernet-bridge`.

5. Enable configuration of global-level mirror destination properties for this address family:

```
[edit forwarding-options port-mirroring family family]
user@host# edit output
```

6. Specify global-level mirror destination properties for this address family.

   a. Specify the physical interface on which to send the mirrored packets:

   ```
   [edit forwarding-options port-mirroring family family output]
   user@host# set interface interface-name
   ```

   You can also specify an integrated routing and bridging (IRB) interface as the output interface.

   b. (Optional) Allow configuration of filters on the destination interface for the named port-mirroring instance:

   ```
   [edit forwarding-options port-mirroring family family output]
   user@host# set no-filter-check
   ```

7. (Optional) Specify that any packets selected for mirroring are to be mirrored only once to any mirroring destination:

```
[edit forwarding-options port-mirroring family family output]
user@host# up 2
[edit forwarding-options port-mirroring]
user@host# set mirror-once
```

TIP: Enable the `mirror-once` option when an MX Series router or an EX Series switch is configured to perform Layer 2 port mirroring at both ingress and egress interfaces, which

could result in sending duplicate packets to the same destination (which would complicate the analysis of the mirrored traffic).

8. Verify the minimum configuration of the global instance of Layer 2 port mirroring:

```
[edit forwarding-options ... ]
user@host# top
[edit]
user@host# show forwarding-options

forwarding-options {
    port-mirroring {
        input { # Global packet-selection properties.
            maximum-packet-length number; # Default is 0.
            rate number;
            run-length number;
        }
        family (ccc | vpls) { # Address- type 'ethernet-switching' displays as 'vpls'.
            output { # Global mirror destination properties.
                interface interface-name;
                no-filter-check; # Optional. Allow filters on interface.
            }
        }
        mirror-once; # Optional. Mirror destinations do not receive duplicate packets.
    }
}
```

Layer 2 Port Mirroring Named Instances

IN THIS SECTION

- Layer 2 Port Mirroring Named Instances Overview | **1128**
- Mirroring at Ports Grouped at the FPC Level | **1128**
- Mirroring at Ports Grouped at the PIC Level | **1129**
- Mirroring at a Group of Ports Bound to Multiple Named Instances | **1129**

This topic describes the following information:

Layer 2 Port Mirroring Named Instances Overview

On an MX Series router and on an EX Series switch, you can define a set of port-mirroring properties that you can explicitly bind to physical ports on the router or switch. This set of *port mirroring* properties is known as a *named instance* of Layer 2 port mirroring.

You can bind a named instance of Layer 2 port mirroring to physical ports associated with an MX Series router's or an EX Series switch's Packet Forwarding Engine components at different levels of the router (or switch) chassis:

- At the FPC level—You can bind a named instance to the physical ports associated with a specific Dense Port Concentrator (DPC) or to the physical ports associated with a specific Flexible Port Concentrator (FPC).
- At the PIC level—You can bind a named instance of port mirroring to a specific Packet Forwarding Engine (on a specific DPC) or to a specific PIC.

> **NOTE**: MX Series routers support DPCs as well as FPCs and PICs. Unlike FPCs, DPCs do not support PICs. In the Junos OS CLI, however, you use FPC and PIC syntax to configure or display information about DPCs and the Packet Forwarding Engines on the DPCs.

The following points summarize the behavior of Layer 2 port mirroring based on named instances:

- The scope of packet selection is determined by the target of the binding—At the ports (or port) bound to a named instance of Layer 2 port mirroring, the router or switch selects input packets according to the packet-selection properties in the named instance.
- The destination of a selected packet is determined by the packet address family—Of the packets selected, the router or switch mirrors only the packets belonging to an address family for which the named instance of Layer 2 port mirroring specifies a set of mirror destination properties. In a Layer 2 environment, MX Series routers and EX Series switches support port mirroring of VPLS (`family ethernet-switching` or `family vpls`) traffic and Layer 2 VPN traffic with `family ccc`.

For a general description of Layer 2 port-mirroring properties, see "Understanding Layer 2 Port Mirroring Properties" on page 1013. For a comparison of the types of Layer 2 port mirroring available on an MX Series router and on an EX Series switch, see Application of Layer 2 Port Mirroring Types.

Mirroring at Ports Grouped at the FPC Level

On an MX Series router and on an EX Series switch, you can bind a named instance of Layer 2 port mirroring to a specific DPC or FPC installed in the router (or switch) chassis. The port mirroring properties in the instance are applied to all Packet Forwarding Engines (and their associated ports) on the specified DPC or to all PICs (and their associated ports) installed in the specified FPC. Port mirroring

properties that are bound to a DPC or FPC override any port-mirroring properties bound at the global level or the MX Series router (or switch) chassis.

Mirroring at Ports Grouped at the PIC Level

On an MX Series router and on an EX Series switch, you can bind a named instance of Layer 2 port mirroring to a specific Packet Forwarding Engine or PIC. The port-mirroring properties in that instance are applied to all ports associated with the specified Packet Forwarding Engine or PIC. Port-mirroring properties that are bound to a Packet Forwarding Engine or PIC override any port-mirroring properties bound at the DPC or FPC that contains them.

NOTE: For MX960 routers, there is a one-to-one mapping of Packet Forwarding Engines to Ethernet ports. Therefore, on MX960 routers only, you can configure port-specific bindings of port-mirroring instances.

Mirroring at a Group of Ports Bound to Multiple Named Instances

On an MX Series router and on an EX Series switch, you can apply up to two named instances of Layer 2 port mirroring to the same group of ports within the router (or switch) chassis. By applying two different port-mirroring instances to the same DPC, FPC, Packet Forwarding Engine, or PIC, you can bind two distinct Layer 2 port mirroring specifications to a single group of ports.

NOTE: You can configure only one global instance of Layer 2 port mirroring on an MX Series router and on an EX Series switch.

NOTE: You can configure more than two port mirroring instances for each FPC by configuring inline port mirroring. For information on inline port mirroring, see "Configuring Inline Port Mirroring" on page 1134.

Defining a Named Instance of Layer 2 Port Mirroring

On an MX Series router and on an EX Series switch, you can define a set of Layer 2 port-mirroring properties that you can bind to a particular Packet Forwarding Engine (at the PIC level of the router or switch chassis) or to a group of Packet Forwarding Engines (at the DPC or FPC level of the chassis).

To define a named instance of Layer 2 port mirroring on an MX Series router or on an EX Series switch:

1. Enable configuration of a named instance of Layer 2 port mirroring :

```
[edit]
user@host# edit forwarding-options port-mirroring instance pm-instance-name
```

2. Enable configuration of the packet-sampling properties:

```
[edit forwarding-options port-mirroring instance pm-instance-name]
user@host# edit input
```

3. Specify packet-selection properties:

   a. Specify the number of packets to select:

   ```
   [edit forwarding-options port-mirroring instance pm-instance-name input]
   user@host# set rate number
   ```

   The valid range is 1 through 65535.

   b. Specify the number of packets to mirror from each selection:

   ```
   [edit forwarding-options port-mirroring instance pm-named-instance input]
   user@host# set run-length number
   ```

   The valid range is 0 through 20. The default value is 0.

   NOTE: The run-length statement is not supported on MX80 routers.

   c. Specify the length to which mirrored packets are to be truncated:

   ```
   [edit forwarding-options port-mirroring instance pm-instance-name input]
   user@host# set maximum-packet-length number
   ```

   The valid range is 0 through 9216. The default value is 0, which means the mirrored packets are not truncated.

   NOTE: The maximum-packet-length statement is not supported on MX80 routers.

4. Enable configuration of the mirror destination properties for Layer 2 packets that are part of bridging domain, Layer 2 switching cross-connects, or virtual private LAN service (VPLS):

   a. Specify the Layer 2 address family type of traffic to be mirrored:

   ```
   [edit forwarding-options port-mirroring instance pm-instance-name input]
   user@host# up
   [edit forwarding-options port-mirroring instance pm-instance-name]
   user@host# edit family family
   ```

   The value of the *family* option can be `ethernet-switching`, `ccc`, or `vpls`.

   > **NOTE**: Under the `[edit forwarding-options port-mirroring]` hierarchy level, the protocol family statement `family ethernet-switching` is an alias for `family vpls`. The command-line interface (CLI) displays Layer 2 port-mirroring configurations as `family vpls`, even for Layer 2 port-mirroring configured as `family ethernet-switching`. Use `family ethernet-switching` when the physical interface is configured with `encapsulation ethernet-bridge`.

   b. Enable configuration of the mirror destination properties:

   ```
   [edit forwarding-options port-mirroring instance pm-instance-name family family]
   user@host# edit output
   ```

5. Specify mirror destination properties.

   a. Specify the physical interface on which to send the mirrored packets:

   ```
   [edit forwarding-options port-mirroring instance pm-instance-name family family output]
   user@host# set interface interface-name
   ```

   b. (Optional) Allow configuration of filters on the destination interface for the global port-mirroring instance:

   ```
   [edit forwarding-options port-mirroring instance pm-instance-name family family output]
   user@host# set no-filter-check
   ```

> **NOTE**: You cannot configure port mirroring instances on MX80 routers. You can only configure port mirroring at the global level on MX80 routers.

6. (Optional) Specify that any packets selected for mirroring are to be mirrored only once to any mirroring destination:

```
[edit forwarding-options port-mirroring instance pm-instance-name family family output]
user@host# up 3
[edit forwarding-options port-mirroring]
user@host# set mirror-once
```

> **TIP**: Enable the global `mirror-once` option when an MX Series router or an EX Series switch is configured to perform Layer 2 port mirroring at both ingress and egress interfaces, which could result in sending duplicate packets to the same destination (which in turn would complicate the analysis of the mirrored traffic).

7. To configure a mirroring destination for a different packet family type, repeat steps 4 through 6.
8. Verify the minimum configuration of the named instances of Layer 2 port mirroring:

```
[edit forwarding-options ... ]
user@host# top
[edit]
user@host# show forwarding-options

forwarding-options {
    port-mirroring {
        ... optional-global-port-mirroring-configuration ...
        instance {
            pm-instance-name ( # A named instance of port mirroring
                input { # Packet-selection properties
                    maximum-packet-length number; # Default is 0.
                    rate number;
                    run-length number;
                }
                family (ccc | vpls) { # Address- type 'ethernet-switching' displays as 'vpls'.
                    output { # Mirror destination properties
                        interface interface-name;
                        no-filter-check; # Optional. Allow filters on interface.
```

```
                    }
                }
            }
        }
        mirror-once; # Optional. Mirror destinations do not receive duplicate packets.
    }
}
```

Disabling Layer 2 Port Mirroring Instances

You can disable the global instance of Layer 2 port mirroring, a particular named instance, or all instances of port mirroring:

- To disable the global instance of Layer 2 port mirroring, include the `disable` statement at the `[edit forwarding-options port-mirroring]` hierarchy level:

```
[edit]
forwarding-options {
    port-mirroring {
        disable; Disables the global instance of Layer 2 port mirroring.
        ...global-instance-of-layer-2-port-mirroring-configuration...
    }
}
```

- To disable the definition of a particular named instance of Layer 2 port mirroring, include the `disable` statement at the `[edit forwarding-options port-mirroring instance` *`instance-name`*`]` hierarchy level:

```
[edit]
forwarding-options {
    port-mirroring {
        ...optional-configuration-of-the-global-instance-of-layer-2-port-mirroring...
        instance {
            port-mirroring-instance-name {
                disable; Disables this named instance of Layer 2 port mirroring.
                ...definition-of-a-named-instance-of-layer-2-port-mirroring...
            }
        }
    }
}
```

- To disable the global instance and all named instances of Layer 2 port mirroring, include the `disable-all-instances` statement at the `[edit forwarding-options port-mirroring]` hierarchy level:

```
[edit]
forwarding-options {
    port-mirroring {
        disable-all-instances; Disables all instances of Layer 2 port mirroring.
        ...optional-configuration-of-the-global-instance-of-layer-2-port-mirroring...
        instance {
            port-mirroring-instance-name {
                ...definition-of-a-named-instance-of-layer-2-port-mirroring...
            }
        }
    }
}
```

Configuring Inline Port Mirroring

Inline port mirroring provides you with the ability to specify instances that are not bound to the flexible PIC concentrator (FPC) in the firewall filter `then port-mirror-instance` action. This way, you are not limited to only two port-mirror instances per FPC. Inline port mirroring decouples the port-mirror destination from the input parameters like `rate`. While the input parameters are programmed in the switch interface board, the next-hop destination of the mirrored packet is available in the packet itself. Inline port mirroring is supported only on Trio-based modular port concentrators (MPCs).

Using inline port mirroring, a port-mirror instance will have an option to inherit input parameters from another instance that specifies it, as shown in the following CLI configuration example:

```
instance pm2 {
    +  input-parameters-instance pm1;
    family inet {
        output {
            interface ge-1/2/3.0 {
                next-hop 192.0.2.10;
            }
        }
    }
}
```

Multiple levels of inheritance are not allowed. One instance can be referred by multiple instances. An instance can refer to another instance that is defined before it. Forward references are not allowed and an instance cannot refer to itself, doing so will cause an error during configuration parsing.

The user can specify an instance that is not bound to the FPC in the firewall filter. The specified filter should inherit one of the two instances that have been bound to the FPC. If it does not, the packet is not marked for port-mirroring. If it does, then the packet will be sampled using the input parameters specified by the referred instance but the copy will be sent to the its own destination.

Configuring Port Mirroring on Physical Interfaces

IN THIS SECTION

- Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface | **1135**
- Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level | **1136**
- Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level | **1137**
- Examples: Layer 2 Port Mirroring at Multiple Levels of the Chassis | **1139**
- Configuring Layer 2 Port Mirroring Over GRE Interface | **1141**
- Example: Configuring Layer 2 Port Mirroring Over a GRE Interface | **1142**

Precedence of Multiple Levels of Layer 2 Port Mirroring on a Physical Interface

You can bind different sets of Layer 2 *port mirroring* properties (the global instance and one or more named instances) at various levels of an MX Series router or of an EX Series switch chassis (at the chassis level, at the FPC level, or at the PIC level). Therefore, it is possible for a single group of physical interfaces to be bound to multiple Layer 2 port mirroring definitions.

If a group of ports (or, in the case of a PIC-level binding in an MX960 router, a single port) is bound to multiple Layer 2 port mirroring definitions, the router (or switch) applies the Layer 2 port-mirroring properties to those ports as follows:

1. Chassis-level port-mirroring properties implicitly apply to all ports in the chassis. If an MX Series router or an EX Series switch is configured with the global port-mirroring instance, those port mirroring properties apply to all ports. See Configuring the Global Instance of Layer 2 Port Mirroring.
2. FPC-level port-mirroring properties override chassis-level properties. If a DPC or FPC is bound to a named instance of port mirroring, those port mirroring properties apply to all ports associated with

that DPC or FPC, overriding any port mirroring properties bound at the chassis level. See "Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level" on page 1136.

3. PIC-level port-mirroring properties override FPC-level properties. If a Packet Forwarding Engine or PIC is bound to a named instance of port-mirroring, those port mirroring properties apply to all ports associated with the Packet Forwarding Engine or PIC, overriding any port-mirroring properties bound to those ports at the FPC level. See "Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level" on page 1137.

Binding Layer 2 Port Mirroring to Ports Grouped at the FPC Level

On an MX Series router and on an EX Series switch, you can bind a named instance of Layer 2 port mirroring to a specific DPC or to a specific FPC in the router (or switch) chassis. This is known as binding a named instance of Layer 2 port mirroring *at the FPC level* of the router (or switch) chassis. The port mirroring properties specified in the named instance are applied to all physical ports associated with all Packet Forwarding Engines on the specified DPC or FPC.

NOTE: You can also bind a named instance of Layer 2 port mirroring to a specific Packet Forwarding Engine on a DPC or FPC in the router (or switch) chassis.

For any packet-type family supported by Layer 2 port mirroring

- Port-mirroring properties bound to a specific DPC or FPC override any port-mirroring properties configured at the global level.
- Port-mirroring properties bound to a specific Packet Forwarding Engine override any port-mirroring properties configured at the DPC or FPC level.

You can apply up to two named instances of Layer 2 port mirroring to the same group of ports within the router (or switch) chassis. By applying two different port-mirroring instances to the same DPC or FPC, you can bind two distinct Layer 2 port-mirroring specifications to a single group of ports.

Before you begin, complete the following tasks:

- Define a named instance of Layer 2 port mirroring. See Defining a Named Instance of Layer 2 Port Mirroring.
- Display information about the number and types of DPCs or FPCs in the MX Series router and in the EX Series switch, the number of Packet Forwarding Engines on each, and the number and types of ports per Packet Forwarding Engine.

To bind a named instance of Layer 2 port mirroring to a DPC or FPC and its Packet Forwarding Engines:

1. Enable configuration of the router (or switch) chassis properties:

```
[edit]
user@host# edit chassis
```

2. Enable configuration of a DPC (and its corresponding Packet Forwarding Engines) or an FPC (and its installed PICs):

```
[edit chassis]
user@host# edit fpc slot-number
```

3. Bind a named instance of Layer 2 port mirroring (*pm-instance-name*) to the DPC or FPC:

```
[edit chassis fpc slot-number]
user@host# set port-mirror-instance pm-instance-name
```

4. (Optional) To bind a second named instance of Layer 2 port mirroring to the same DPC or FPC, repeat the previous step (step 3) and specify a different named instance of Layer 2 port mirroring.
5. Verify the minimum configuration of the binding:

```
[edit chassis fpc slot-number port-mirror-instance pm-instance-name]
user@host# top
[edit]
user@host# show chassis

chassis {
    fpc slot-number { # Bind two port mirroring named instances at the FPC level.
        port-mirror-instance pm-instance-name-1;
        port-mirror-instance pm-instance-name-2;
    }
}
```

Binding Layer 2 Port Mirroring to Ports Grouped at the PIC Level

On an MX Series router and on an EX Series switch, you can bind a named instance of Layer 2 port mirroring to the ports associated with a specific Packet Forwarding Engine (on a DPC) or to the ports associated with a specific PIC (installed in an FPC). This is known as binding a named instance of Layer 2 port mirroring *at the PIC level* of the router (or switch) chassis. The port-mirroring properties specified in the named instance are applied to all physical ports associated with the specified Packet Forwarding Engine.

NOTE: You can also bind a named instance of Layer 2 port mirroring to a specific DPC or FPC in the router (or switch) chassis.

For any packet-type family supported by Layer 2 port mirroring:

- Port-mirroring properties bound to a specific Packet Forwarding Engine override any port-mirroring properties configured at the DPC or FPC level.
- Port-mirroring properties bound to a specific DPC or FPC override any port-mirroring properties configured at the global level.

You can apply up to two named instances of Layer 2 port-mirroring to the same group of ports within the router (or switch) chassis. By applying two different port-mirroring instances to the same Packet Forwarding Engine or PIC, you can bind two distinct Layer 2 port mirroring specifications to a single group of ports.

For MX960 routers, there is a one-to-one mapping of Packet Forwarding Engines to Ethernet ports. Therefore, on MX960 routers only, you can bind a named instance of Layer 2 port mirroring to a *specific port* by binding the instance to the Packet Forwarding Engine associated with the port.

Before you begin, complete the following tasks:

- Define a named instance of Layer 2 port mirroring. See Defining a Named Instance of Layer 2 Port Mirroring.
- Display information about the number and types of DPCs in the MX Series router or in the EX Series switch, the number of Packet Forwarding Engines on each DPC, and the number and types of ports per Packet Forwarding Engine.

To bind a named instance of Layer 2 port mirroring to a Packet Forwarding Engine:

1. Enable configuration of the router (or switch) chassis properties:

```
[edit]
user@host# edit chassis
```

2. Enable configuration of a Packet Forwarding Engine or PIC:

```
[edit chassis]
user@host# edit fpc slot-number
user@host# edit pic slot-number
```

3. Bind a named instance of Layer 2 port mirroring (*pm-instance-name*) to the Packet Forwarding Engine or PIC:

```
[edit chassis fpc slot-number pic slot-number]
user@host# set port-mirror-instance pm-instance-name
```

4. (Optional) To bind a second named instance of Layer 2 port mirroring to the same Packet Forwarding Engine or PIC, repeat the previous step (step 3) and specify a different named instance of Layer 2 port mirroring.
5. Verify the minimum configuration of the binding:

```
[edit forwarding-options ... ]
user@host# top
[edit]
user@host# show chassis
chassis {
    fpc slot-number {
        ... optional-binding-of-a-port-mirroring-instance-at-the-dpc-level ...
        pic slot-number { # Bind two port-mirroring named instances at the PIC level.
            port-mirror-instance pm-instance-name-1;
            port-mirror-instance pm-instance-name-2;
        }
    }
}
```

Examples: Layer 2 Port Mirroring at Multiple Levels of the Chassis

IN THIS SECTION

- Layer 2 Port Mirroring at the FPC Level | **1140**
- Layer 2 Port Mirroring at the PIC Level | **1140**
- Layer 2 Port Mirroring at the FPC and PIC Levels | **1141**

On an MX Series router or on an EX Series switch, you can apply named instances of Layer 2 port mirroring at the FPC or DPC level of the chassis or at the PIC level of the chassis. However, you can configure (and implicitly apply) only one global instance of Layer 2 port mirroring to the entire chassis.

Layer 2 Port Mirroring at the FPC Level

In this example configuration of an MX Series router or of an EX Series switch chassis, a named instance of Layer 2 port mirroring (**pm1**) is bound to physical ports grouped at the FPC level:

```
[edit]
chassis {
    fpc 2 {
        port-mirror-instance pm1;
    }
}
```

This is not a complete configuration. The physical interfaces associated with the FPC or DPC in slot 2 must be configured at the `[edit interfaces]` hierarchy level. The Layer 2 port mirroring named instance **pm1** must be configured at the `[edit forwarding-options port-mirroring instance]` hierarchy level.

Layer 2 Port Mirroring at the PIC Level

In this example configuration of an MX Series router or of an EX Series switch chassis, a named instance of Layer 2 port mirroring (**pm2**) is bound to the physical ports grouped at the PIC level:

```
[edit]
chassis {
    fpc 2 {
        pic 0 {
            port-mirror-instance pm2;
        }
    }
}
```

This is not a complete configuration. The physical interfaces associated with the FPC or DPC in slot 2 must be configured at the `[edit interfaces]` hierarchy level. The Layer 2 port mirroring named instance **pm2** must be configured at the `[edit forwarding-options port-mirroring instance]` hierarchy level.

Layer 2 Port Mirroring at the FPC and PIC Levels

In this example configuration of an MX Series router chassis or an EX Series switch, one named instance of Layer 2 port mirroring (**pm1**) is applied at the FPC level of the router (or switch) chassis. A second named instance (**pm2**) is applied at the PIC level:

```
[edit]
chassis {
    fpc 2 {
        port-mirror-instance pm1;
        pic 0 {
            port-mirror-instance pm2;
        }
    }
}
```

This is not a complete configuration. Physical interfaces associated with the FPC or DPC in slot 2, including physical interfaces associated with **pic 0**, must be configured at the `[edit interfaces]` hierarchy level. The Layer 2 port mirroring named instances **pm1** and **pm2** must be configured at the `[edit forwarding-options port-mirroring instance]` hierarchy level.

Configuring Layer 2 Port Mirroring Over GRE Interface

Port mirroring is the ability of a router to send a copy of a packet to an external host address or a packet analyzer for analysis. One application for port mirroring sends a duplicate packet to a virtual tunnel. A next-hop group can then be configured to forward copies of this duplicate packet to several interfaces. Junos OS supports Layer 2 port mirroring to a remote collector over a GRE interface.

To configure layer 2 port-mirroring over a GRE interface, do the following:

1. Configure the GRE interface with the source and destination address.

```
[edit interfaces inteface-name unit unit-number tunnel]
set source ip-address
set  destination ip-address
```

2. Configure family bridge parameters on the GRE interface.

```
[edit interfaces interface-name unit unit-number family bridge]
set interface-mode trunk
set vlan-id valn-id
```

3. Configure the rate at which the input packets are mirrored.

```
[edit forwarding-options port-mirroring]
set f input rate rate
```

4. Configure the output interface for family VPLS for the GRE interface.

```
[edit forwarding-options family vpls]
set output  interface gre-interface-name
```

5. Configure the firewall filter term for family bridge to count packets arriving at the interface.

```
[edit firewall family bridge]
set filter f1 term term then count count
```

6. Configure firewall filter term for family bridge to mirror the packets.

```
[edit firewall family bridge]
set filter filter-name term term then port-mirror
```

SEE ALSO

Tunnel Services Overview

Example: Configuring Layer 2 Port Mirroring Over a GRE Interface

IN THIS SECTION

- Requirements | **1143**
- Overview | **1143**
- Configuration | **1144**
- Verification | **1149**

This example shows how to configure Layer 2 port mirroring over a GRE interface for analysis.

Requirements

This example uses the following hardware and software components:

- One MX Series router
- Junos OS Release 16.1 or later running on all devices

Overview

IN THIS SECTION

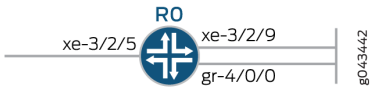
- Topology | 1143

Port mirroring is the ability of a router to send a copy of a packet to an external host address or a packet analyzer for analysis. One application for port mirroring sends a duplicate packet to a virtual tunnel. A next-hop group can then be configured to forward copies of this duplicate packet to several interfaces. Starting with Junos OS Release 16.1, Layer 2 port mirroring to a remote collector over a GRE interface is supported.

Topology

Figure 41 on page 1143 shows port mirroring configured over a GRE interface. The interface gr-4/0/0 is configured as family bridge. Firewall family bridge filter f1 is configured as port-mirror. Mirror destination is configured as gr-4/0/0. Firewall family bridge filter f1 is applied at the ingress and egress of the xe-3/2/5.0 interface, which mirrors packets to mirror destination gr-4/0/0.

Figure 41: Example Layer 2 Port Mirroring over GRE Interface



Configuration

IN THIS SECTION

- CLI Quick Configuration | 1144
- Configuring R0 | 1145
- Results | 1146

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the **[edit]** hierarchy level, and then enter `commit` from configuration mode.

R0

```
set chassis fpc4 pic0 tunnel-services bandwidth 10g
set chassis network-services enhanced-ip
set interfaces xe-3/2/5 flexible-vlan-tagging
set interfaces xe-3/2/5 encapsulation flexible-ethernet-services
set interfaces xe-3/2/5  unit 0 encapsulation vlan-bridge
set interfaces xe-3/2/5 unit 0 vlan-id 100
set interfaces xe-3/2/5 unit 0 family bridge filter  input f1
set interfaces xe-3/2/5 unit 0 family bridge filter output f1
set interfaces xe-3/2/9 flexible-vlan-tagging
set interfaces xe-3/2/9 encapsulation flexible-ethernet-services
set interfaces xe-3/2/9  unit 0 encapsulation vlan-bridge
set interfaces xe-3/2/9  unit 0 vlan-id 100
set interfaces gr-4/0/0  unit 0 tunnel source 10.1.1.1
set interfaces gr-4/0/0  unit 0 tunnel  destination 10.1.1.2
set interfaces gr-4/0/0  unit 0  family bridge  interface-mode trunk
set interfaces gr-4/0/0  unit 0  family bridge   vlan-id 100
set forwarding-options port-mirroring input rate 1
set forwarding-options  family vpls output  interface gr-4/0/0.0
set firewall family bridge  filter f1 term t  then count c
set firewall family bridge  filter f1 term t  then port-mirror
set bridge-domains  b vlan-id 100
```

```
set bridge-domains  b  interface xe-3/2/5.0
set bridge-domains  b  interface xe-3/2/9.0
```

Configuring R0

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For information about navigating the CLI, see "*Using the CLI Editor in Configuration Mode*" in the *Junos OS CLI User Guide* .

To configure Device R0:

1. Configure the flexible PIC concentrator parameters of the chassis.

```
[edit chassis]
user@R0# set fpc4 pic0 tunnel-services bandwidth 10g
user@R0# set network-services enhanced-ip
```

2. Configure the enhanced-ip network services of the chassis.

```
[edit chassis]
user@R0# set network-services enhanced-ip
```

3. Configure the interfaces.

```
[edit interfaces]
user@R0# set xe-3/2/5 flexible-vlan-tagging
user@R0# set xe-3/2/5 encapsulation flexible-ethernet-services
user@R0# set xe-3/2/5  unit 0 encapsulation vlan-bridge
user@R0# set xe-3/2/5 unit 0 vlan-id 100
user@R0# set xe-3/2/5 unit 0 family bridge filter  input f1
user@R0# set xe-3/2/5 unit 0 family bridge filter output f1
user@R0# set xe-3/2/9 flexible-vlan-tagging
user@R0# set xe-3/2/9 encapsulation flexible-ethernet-services
user@R0# set xe-3/2/9  unit 0 encapsulation vlan-bridge
user@R0# set xe-3/2/9  unit 0 vlan-id 100
user@R0# set gr-4/0/0  unit 0 tunnel source 10.1.1.1
user@R0# set gr-4/0/0  unit 0 tunnel  destination 10.1.1.2
```

```
user@R0# set gr-4/0/0  unit 0  family bridge  interface-mode trunk
user@R0# set gr-4/0/0  unit 0  family bridge   vlan-id 100
```

4. Configure the rate of input packets to be sampled.

```
[edit forwarding-options]
user@R0# set port-mirroring input rate 1
```

5. Configure the output interface for the VPLS address family of packets to mirror.

```
[edit forwarding-options]
user@R0# set family vpls output  interface gr-4/0/0.0
```

6. Configure the protocol family BRIDGE for the firewall filter.

```
[edit firewall]
user@R0# set family bridge  filter f1 term t  then count c
user@R0# set family bridge  filter f1 term t  then port-mirror
```

7. Configure the VLAN ID for the bridge domain.

```
[edit bridge-domains]
user@R0# set  b vlan-id 100
user@R0# set  b  interface xe-3/2/5.0
user@R0# set  b  interface xe-3/2/9.0
```

8. Configure the interface for the bridge domain.

```
[edit bridge-domains]
user@R0# set  b  interface xe-3/2/5.0
user@R0# set  b  interface xe-3/2/9.0
```

Results

From configuration mode, confirm your configuration by entering the **show bridge-domains**, **show chassis**, **show forwarding-options**, **show firewall**, and **show interfaces** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@R0# show chassis
fpc 4 {
    pic 0 {
        tunnel-services {
            bandwidth 10g;
        }
    }
}
network-services enhanced-ip;
```

```
user@R0# show interfaces
}
xe-3/2/5 {
    flexible-vlan-tagging;
    encapsulation flexible-ethernet-services;
    unit 0 {
        encapsulation vlan-bridge;
        vlan-id 100;
        family bridge {
            filter {
                input f1;
                output f1;
            }
        }
    }
}
xe-3/2/9 {
    flexible-vlan-tagging;
    encapsulation flexible-ethernet-services;
    unit 0 {
        encapsulation vlan-bridge;
        vlan-id 100;
    }
}
```

```
gr-4/0/0 {
    unit 0 {
        tunnel {
            source 10.1.1.1;
            destination 10.1.1.2;
        }
        family bridge {
            interface-mode trunk;
            vlan-id 100;
        }
    }
}
```

```
user@R0# show forwarding-options
port-mirroring {
    input {
        rate 1;
    }
    family vpls {
        output {
            interface gr-4/0/0.0;
        }
    }
}
```

```
user@R0# show firewall
family bridge {
    filter f1 {
        term t {
            then {
                count c;
                port-mirror;
            }
        }
    }
}
```

```
user@R0# show bridge-domains
b {
```

```
    vlan-id 100;
    interface xe-3/2/5.0;
    interface xe-3/2/9.0;
}
```

Verification

IN THIS SECTION

- Verifying Port Mirroring of Traffic | **1149**

Confirm that the configuration is working properly.

Verifying Port Mirroring of Traffic

Purpose

Display port mirroring of traffic information.

Action

On Device R0, from operational mode, run the `show forwarding-options port-mirroring` command to display the port mirroring of traffic information.

```
user@R0> show forwarding-options port-mirroring
Instance Name: & globalinstance
  Instance Id: 1
  Input parameters:
    Rate                  : 1
    Run-length            : 0
    Maximum-packet-length : 0
  Output parameters:
    Family              State     Destination          Next-hop
    vpls                up        gr-4/0/0.0

Instance Name: pm_instance
  Instance Id: 2
```

```
Input parameters:
  Rate                 : 10
  Run-length           : 0
  Maximum-packet-length : 0
Output parameters:
  Family              State     Destination          Next-hop
  vpls                up        gr-4/0/0.0
```

Meaning

The output shows the port mirroring of traffic information.

Configuring Port Mirroring on Logical Interfaces

IN THIS SECTION

- Layer 2 Port Mirroring Firewall Filters | **1151**
- Defining a Layer 2 Port-Mirroring Firewall Filter | **1153**
- Configuring Protocol-Independent Firewall Filter for Port Mirroring | **1156**
- Example: Mirroring Employee Web Traffic with a Firewall Filter | **1158**
- Layer 2 Port Mirroring of PE Router or PE Switch Logical Interfaces | **1164**
- Layer 2 Port Mirroring of PE Router or PE Switch Aggregated Ethernet Interfaces | **1166**
- Applying Layer 2 Port Mirroring to a Logical Interface | **1166**
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain | **1169**
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance | **1171**
- Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VLAN | **1174**
- Example: Layer 2 Port Mirroring at a Logical Interface | **1176**
- Example: Layer 2 Port Mirroring for a Layer 2 VPN | **1179**
- Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links | **1182**

Layer 2 Port Mirroring Firewall Filters

IN THIS SECTION

- Layer 2 Port Mirroring Firewall Filters Overview | **1151**
- Mirroring of Packets Received or Sent on a Logical Interface | **1152**
- Mirroring of Packets Forwarded or Flooded to a VLAN | **1152**
- Mirroring of Packets Forwarded or Flooded to a VPLS Routing Instance | **1153**

This topic describes the following information:

Layer 2 Port Mirroring Firewall Filters Overview

On an MX Series router and on an EX Series switch, you can configure a *firewall filter term* to specify that Layer 2 *port mirroring* is to be applied to all packets at the interface to which the firewall filter is applied.

You can apply a Layer 2 port-mirroring firewall filter to the input or output logical interfaces (including aggregated Ethernet logical interfaces), to traffic forwarded or flooded to a VLAN, or traffic forwarded or flooded to a VPLS routing instance.

MX Series routers and EX Series switches support Layer 2 port mirroring of VPLS (`family ethernet-switching` or `family vpls`) traffic and Layer 2 VPN traffic with `family ccc` in a Layer 2 environment

Within a firewall filter `term`, you can specify the Layer 2 port-mirroring properties under the `then` statement in either of the following ways:

- Implicitly reference the Layer 2 port mirroring properties in effect on the port.
- Explicitly reference a particular named instance of Layer 2 port mirroring.

NOTE: When configuring a Layer 2 port-mirroring firewall filter, do not include the optional `from` statement that specifies match conditions based on the route source address. Omit this statement so that all packets are considered to match and all *actions* and *action-modifiers* specified in the `then` statement are taken.

If you want to mirror all incoming packets, then you must not use the from statement; /* comment: one configure filter terms with from if they are interested in mirroring only a subset of packets.

NOTE: If you associate integrated routing and bridging (IRB) with the VLAN (or VPLS routing instance), and also configure within the VLAN (or VPLS routing instance) a forwarding table filter with the `port-mirror` or `port-mirror-instance` action, then the IRB packet is mirrored as a Layer 2 packet. You can disable this behavior by configuring the no-irb-layer-2-copy statement in the VLAN (or VPLS routing instance).

For a detailed description of how to configure a Layer 2 port-mirroring firewall filter, see Defining a Layer 2 Port-Mirroring Firewall Filter.

For detailed information about how you can use Layer 2 port-mirroring firewall filters with MX Routers and EX Series switches configured as provider edge (PE) routers or PE switches, see Understanding Layer 2 Port Mirroring of PE Router Logical Interfaces. For detailed information about configuring firewall filters in general (including in a Layer 3 environment), see the Routing Policies, Firewall Filters, and Traffic Policers User Guide.

Mirroring of Packets Received or Sent on a Logical Interface

To mirror Layer 2 traffic received or sent on a *logical interface*, apply a port-mirroring firewall filter to the input or output of the interface.

A port-mirroring firewall filter can also be applied to an aggregated-Ethernet logical interface. For details, see Understanding Layer 2 Port Mirroring of PE Router Aggregated Ethernet Interfaces.

NOTE: If port-mirroring firewall filters are applied at both the input and output of a logical interface, two copies of each packet are mirrored. To prevent the router or switch from forwarding duplicate packets to the same destination, you can enable the “mirror-once” option for Layer 2 port mirroring in the global instance for the Layer 2 packet address family.

Mirroring of Packets Forwarded or Flooded to a VLAN

To mirror Layer 2 traffic forwarded to or flooded to a VLAN, apply a port-mirroring firewall filter to the input to the forwarding table or flood table. Any packet received for the VLAN forwarding or flood table and that matches the filter conditions is mirrored.

For more information about VLANs, see Understanding Layer 2 Bridge Domains . For information about flooding behavior in a VLAN, see Understanding Layer 2 Learning and Forwarding for Bridge Domains .

NOTE: When you configure port mirroring on any interface under one VLAN, the mirrored packet can move to an external analyzer located on different VLANs.

Mirroring of Packets Forwarded or Flooded to a VPLS Routing Instance

To mirror Layer 2 traffic forwarded to or flooded to a VPLS routing instance, apply a port-mirroring firewall filter to the input to the forwarding table or flood table. Any packet received for the VPLS routing instance forwarding or flood table and that matches the filter condition is mirrored.

For more information about VPLS routing instances, see *Configuring a VPLS Routing Instance* and Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances. For information about flooding behavior in VPLS, see the Junos OS VPNs Library for Routing Devices.

Defining a Layer 2 Port-Mirroring Firewall Filter

For virtual private LAN service (VPLS) traffic (`family ethernet-switching` or `family vpls`) and for Layer 2 VPNs with family `ccc`on MX Series routers and on EX Series switches only, you can define a firewall filter that specifies Layer 2 port mirroring as the action to be performed if a packet matches the conditions configured in the firewall filter term.

You can use a Layer 2 port-mirroring firewall filter in the following ways:

- To mirror packets received or sent on a logical interface.
- To mirror packets forwarded or flooded to a VLAN.
- To mirror packets forwarded or flooded to a VPLS routing instance.
- To mirror tunnel interface input packets only to multiple destinations.

For a summary of the three types of Layer 2 port-mirroring you can configure on an MX Series router and on an EX Series switch, see Application of Layer 2 Port Mirroring Types.

To define a firewall filter with a Layer 2 port-mirroring action:

1. Enable configuration of firewall filters for Layer 2 packets that are part of a VLAN, a Layer 2 switching cross-connect, or a virtual private LAN service (VPLS):

   ```
   [edit]
   user@host# edit firewall family family
   ```

   The value of the `family` option can be `ethernet-switching` , `ccc`, or `vpls`.

2. Enable configuration of a firewall filter `pm-filter-name`:

```
[edit firewall family family]
user@host# edit filter pm-filter-name
```

3. Enable configuration of a firewall filter term `pm-filter-term-name`:

```
[edit firewall family family filter pm-filter-name]
user@host# edit term pm-filter-term-name
```

4. (Optional) Specify the firewall filter match conditions based on the route source address *only if* you want to mirror a subset of the sampled packets.

   - For detailed information about Layer 2 bridging firewall filter match conditions (which are supported on MX Series routers and EX Series switches only), see *Firewall Filter Match Conditions for Layer 2 Bridging Traffic*.
   - For detailed information about VPLS firewall filter match conditions, see *Firewall Filter Match Conditions for VPLS Traffic*.
   - For detailed information about Layer 2 circuit cross-connect (CCC) firewall filter match conditions, see *Firewall Filter Match Conditions for Layer 2 CCC Traffic*.

   NOTE: If you want all sampled packets to be considered to match (and be subjected to the actions specified in the `then` statement), then omit the `from` statement altogether.

5. Enable configuration of the `action` and `action-modifier` to apply to matching packets:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name]
user@host# edit then
```

6. Specify the actions to be taken on matching packets:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set action
```

   The recommended value for the `action` is `accept`. If you do not specify an action, or if you omit the `then` statement entirely, all packets that match the conditions in the `from` statement are accepted.

7. Specify Layer 2 port mirroring or a next-hop group as the `action-modifier`:

- To reference the Layer 2 port mirroring properties currently in effect for the Packet Forwarding Engine or PIC associated with the underlying physical interface, use the `port-mirror` statement:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set port-mirror
```

- To reference the Layer 2 port mirroring properties configured in a specific named instance, use the port-mirror-instance *pm-instance-name* action modifier:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set port-mirror-instance pm-instance-name
```

  If the underlying physical interface is not bound to a named instance of Layer 2 port mirroring but instead is implicitly bound to the global instance of Layer 2 port mirroring, then traffic at the logical interface is mirrored according to the properties specified in the named instance referenced by the `port-mirror-instance` action modifier.

- To reference a next-hop group that specifies the next-hop addresses (for sending additional copies of packets to an analyzer), use the `next-hop-group` *pm-next-hop-group-name* action modifier:

```
[edit firewall family family filter pm-filter-name term pm-filter-term-name then]
user@host# set next-hop-group pm-next-hop-group-name
```

  For configuration information about next-hop groups, see Defining a Next-Hop Group for Layer 2 Port Mirroring. If you specify a next-hop group for Layer 2 port mirroring, the firewall filter term applies to the tunnel interface input only.

8. Verify the minimum configuration of the Layer 2 port-mirroring firewall filter:

```
[edit firewall ... ]
user@host# top
[edit]
user@host# show firewall

family (ethernet-switching | ccc | vpls) { # Type of packets to mirror
    filter pm-filter-name { # Firewall filter name
        term pm-filter-term-name {
            from { # Do not specify match conditions based on route source address
            }
            then {
                action; # Recommended action is 'accept'
```

```
                action-modifier; # Three options for Layer 2 port mirroring
            }
        }
    }
}
```

In the firewall filter term `then` statement, the *action-modifier* can be `port-mirror`, `port-mirror-instance` , or `next-hop-group` *pm-next-hop-group-name*.

Configuring Protocol-Independent Firewall Filter for Port Mirroring

On MX Series routers with MPCs, you can configure a firewall filter to mirror Layer 2 and Layer 3 packets at a global level and at an instance level. When port mirror is configured at ingress or egress, the packet entering or exiting an interface is copied and the copies are sent to the local interface for local monitoring.

NOTE: Starting with Junos OS Release 13.3R6, only MPC interfaces support `family any` to do port mirroring. DPC interfaces do not support `family any`.

Typically, the firewall filter is configured such that it mirrors either Layer 2 or Layer 3 packets based on the family configured at the interface. However, in case of an integrated routing and bridging (IRB) interface, Layer 2 packets are not completely mirrored because IRB interfaces are configured to mirror only Layer 3 packets. On such an interface, you can configure a firewall filter and port mirroring parameters in the family **any** to ensure that a packet is completely mirrored irrespective of whether it is a Layer 2 or a Layer 3 packet.

NOTE:

- For port mirroring at an instance, you can configure one or more families such as **inet**, **inet6**, **ccc**, and **vpls** simultaneously for the same instance.
- In case of Layer 2 port mirroring, VLAN tags, MPLS headers are retained and can be seen in the mirrored copy at egress.
- For VLAN normalization, the information before normalization is seen for a mirrored packet at ingress. Similarly, at egress, the information after normalization is seen for the mirrored packet.

Before you begin configuring port mirroring, you must configure valid physical interfaces.

To configure a protocol-independent firewall filter for port mirroring:

1. Configure a global firewall filter for mirroring egress or ingress traffic.

```
[edit firewall family any]
user@host# set filter filter-name {
    term term-name {
        then {
            port-mirror;
            accept;
        }
    }
}
```

2. Configure a firewall filter to mirror traffic for an instance.

```
[edit firewall family any]
user@host# set filter filter-name {
    term term-name {
        then {
            port-mirror-instance instance-name;
            accept;
        }
    }
}
```

3. Configure mirroring parameters for egress and ingress traffic.

```
[edit forwarding-options port-mirroring]
user@host# input {
    maximum-packet-length bytes
    rate rate;
}
family any {
    output {
        (next-hop-group group-name | interface interface-name);
    }
}
```

4. Configure mirroring parameters for an instance. In this configuration, you can specify the output or destination for the Layer 2 packets to be either a valid next-hop group or a Layer 2 interface.

```
[edit forwarding-options port-mirroring]
user@host#instance instance-name {
    family any{
        output {
            (next-hop-group group-name | interface interface-name);
        }
    }
}
```

5. Configure the firewall filter at the ingress or egress interface on which the packets are transmitted.

```
[edit interface interface-name unit]
user@host# filter {
    output filter-name;
    input filter-name;
}
```

Example: Mirroring Employee Web Traffic with a Firewall Filter

IN THIS SECTION

- Requirements | **1158**
- Overview | **1159**
- Configuring | **1160**
- Verification | **1163**

Requirements

This example uses the following hardware and software components:

- One switch
- Junos 14.1X53-D20

Overview

IN THIS SECTION

- Topology | 1159

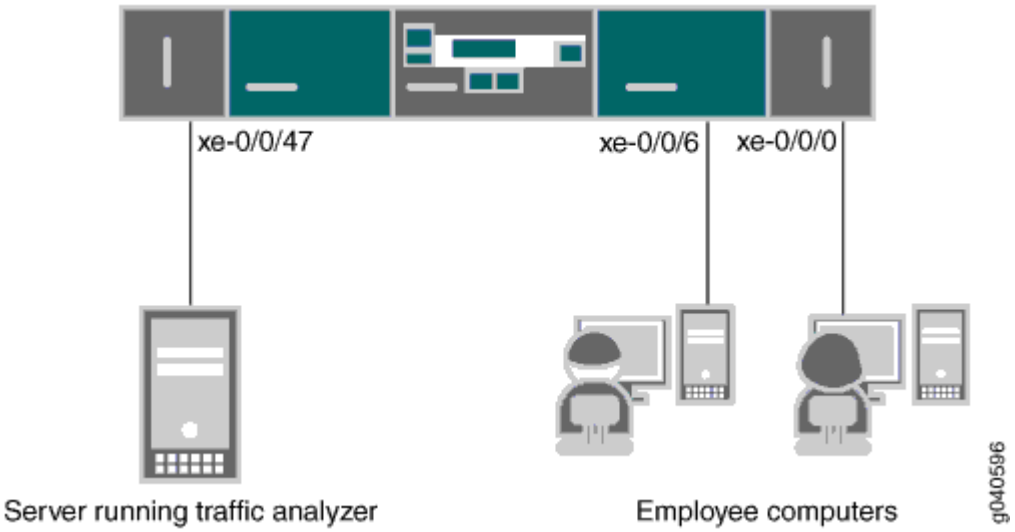
In this example, `xe-0/0/0` and `xe-0/0/6` serve as connections for employee computers. Interface `xe-0/0/47` is connected to a device running an analyzer application.

Rather than mirror all traffic, it is usually desirable to mirror only certain traffic. This is a more-efficient use of your bandwidth and hardware and might be necessary because of constraints on these assets. This example mirrors only traffic sent from employee computers to the Web.

Topology

Figure 42 on page 1159 shows the network topology for this example.

Figure 42: Network Topology for Local Port Mirroring Example



Configuring

IN THIS SECTION

- Procedure | 1160

To specify that the only traffic that will be mirrored is traffic sent by employees to the Web, perform the tasks explained in this section. To select this traffic for mirroring, you use a firewall filter to specify this traffic and direct it to a port-mirroring instance.

Procedure

CLI Quick Configuration

To quickly configure local port mirroring of traffic from employee computers that is destined for the Web, copy the following commands and paste them into a switch terminal window:

```
[edit]
 set forwarding-options port-mirroring family inet output interface xe-0/0/47.0 next-hop
192.0.2.100/24
 set firewall family inet filter watch-employee term employee-to-corp from destination-address
192.0.2.16/24
 set firewall family inet filter watch-employee term employee-to-corp from source-address
192.0.2.16/24
 set firewall family inet filter watch-employee term employee-to-corp then accept
 set firewall family inet filter watch-employee term employee-to-web from destination-port 80
 set firewall family inet filter watch-employee term employee-to-web then port-mirror
 set interfaces xe-0/0/0 unit 0 family address 192.0.1.1/24
 set interfaces xe-0/0/6 unit 0 family address 192.0.1.2/24
 set interfaces xe-0/0/47 unit 0 family address 192.0.1.3/24
 set interfaces xe-0/0/0 unit 0 family inet filter input watch-employee
 set interfaces xe-0/0/6 unit 0 family inet filter input watch-employee
```

Step-by-Step Procedure

To configure local port mirroring of employee to web traffic from the two ports connected to employee computers:

1. Configure a port-mirroring instance, including the output interface and the IP address of the device running the analyzer application as the next hop. (Configure only the output—the input comes from the filter.) You must also specifying that the mirror is for IPv4 traffic (`family inet`).

```
[edit forwarding-options]
user@switch# set forwarding-options port-mirroring family inet output interface xe-0/0/47.0
next-hop 192.0.2.100/28
```

2. Configure an IPv4 (`family inet`) firewall filter called `watch-employee` that includes a term to match traffic sent to the Web and send it to the port-mirroring instance. Traffic sent to and arriving from the corporate subnet (destination or source address of `192.0.nn.nn/24`) does not need to be copied, so first create another term to accept that traffic before it reaches the term that sends Web traffic to the instance:

```
[edit firewall family inet]
er@switch# set filter watch-employee term employee-to-corp from destination-address
192.0.nn.nn/24
user@switch# set filter watch-employee term employee-to-corp from source-address
192.0.nn.nn/24
user@switch# set filter watch-employee term employee-to-corp then accept
user@switch# set filter watch-employee term employee-to-web from destination-port 80
user@switch# set filter watch-employee term employee-to-web then port-mirror
```

3. Configure addresses for the IPv4 interfaces connected to the employee computers and the analyzer device:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family inet address 192.0.1.1/24
user@switch# set xe-0/0/6 unit 0 family inet address 192.0.1.2/24
user@switch# set interfaces xe-0/0/47 unit 0 family address 192.0.1.3/24
```

4. Apply the firewall filter to the appropriate interfaces as an ingress filter:

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family inet filter input watch-employee
user@switch# set xe-0/0/6 unit 0 family inet filter input watch-employee
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show
forwarding-options {
    port-mirroring {
            employee-web-monitor    {
                output {
                    ip-address 192.0.2.100.0;
                    }
                }
            }
        }
    }
}
...
firewall family inet {
    filter watch-employee {
        term employee-to-corp {
            from {
                destination-address 192.0.2.16/24;
                source-address 192.0.2.16/24;
            }
            then accept {
        }
        term employee-to-web {
            from {
                destination-port 80;
            }
            then port-mirror;
        }
    }
}
...
interfaces {
    xe-0/0/0 {
        unit 0 {
            family inet {
                filter {
                    input watch-employee;
```

```
                }
            }
        }
    }
    xe-0/0/6 {
        family inet {
            filter {
                input watch-employee;
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1163**

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer has been created on the switch with the appropriate input interfaces and appropriate output interface.

Action

You can verify that the port mirror analyzer has been configured as expected using the `show forwarding-options port-mirroring` command.

```
user@switch> show forwarding-options port-mirroring
Instance Name: &global_instance
  Instance Id: 1
  Input parameters:
    Rate                 : 1
    Run-length           : 0
    Maximum-packet-length : 0
```

```
Output parameters:
  Family              State     Destination          Next-hop
  inet                up      xe-0/0/47.0          192.0.2.100
```

Meaning

This output shows that the port-mirroring instance has a ratio of 1 (mirroring every packet, the default setting) and the maximum size of the original packet that was mirrored (`0` indicates the entire packet). If the state of the output interface is down or if the output interface is not configured, the value of state will be `down` and the instance will not be programmed for mirroring.

Layer 2 Port Mirroring of PE Router or PE Switch Logical Interfaces

For a router or switch configured as a provider edge (PE) device on the customer-facing edge of a service provider network, you can apply a Layer 2 port-mirroring *firewall filter* at the following ingress and egress points to mirror the traffic between the router or switch and customer edge (CE) devices, which are typically also routers and Ethernet switches.

Table 123 on page 1164 describes the ways in which you can apply Layer 2 port-mirroring firewall filters to a router or switch configured as a PE device.

Table 123: Application of Layer 2 Port Mirroring Firewall Filters on PE Devices

Point of Application	Scope of Mirroring	Notes	Configuration Details
Ingress Customer-Facing Logical Interface	Packets originating within a service provider customer's network, sent first to a CE device, and sent next to the PE device.	You can also configure aggregated Ethernet interfaces between CE devices and PE devices for VPLS routing instances. Traffic is load-balanced across all of the links in the aggregated interface. Traffic received on an aggregated Ethernet interface is forwarded over a different interface based on a lookup of the destination MAC (DMAC) address:	See Applying Layer 2 Port Mirroring to a Logical Interface. For more information about VPLS routing instances, see *Configuring a VPLS Routing Instance* and Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances.

Table 123: Application of Layer 2 Port Mirroring Firewall Filters on PE Devices *(Continued)*

Point of Application	Scope of Mirroring	Notes	Configuration Details
Egress Customer-Facing Logical Interface	Unicast packets being forwarded by the PE device to another PE device. **NOTE:** If you apply a port-mirroring filter to the output for a *logical interface*, only unicast packets are mirrored. To mirror multicast, unknown unicast, and broadcast packets, apply a filter to the input to the flood table of a VLAN or VPLS routing instance.	• Packets destined for a local site are sent out of the load-balanced child interface. • Packets destined for the remote site are encapsulated and forwarded over a label-switched path (LSP).	See Applying Layer 2 Port Mirroring to a Logical Interface.
Input to a VLAN Forwarding Table or Flood Table	Forwarding traffic or flood traffic sent to the VLAN from a CE device.	Forwarding and flood traffic typically consists of broadcast packets, multicast packets, unicast packets with an unknown destination MAC address, or packets with a MAC entry in the DMAC routing table.	See "Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain" on page 1169. For information about flooding behavior in VPLS, see the Junos OS VPNs Library for Routing Devices.
Input to a VPLS Routing Instance Forwarding Table or Flood Table	Forwarding traffic or flood traffic sent to the VPLS routing instance from a CE device.		See Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance. For information about flooding behavior in VPLS, see the Junos OS VPNs Library for Routing Devices.

Layer 2 Port Mirroring of PE Router or PE Switch Aggregated Ethernet Interfaces

An aggregated Ethernet interface is a virtual aggregated link that consists of a set of physical interfaces of the same speed and operating in full-duplex link connection mode. You can configure aggregated Ethernet interfaces between CE devices and PE devices for VPLS routing instances. Traffic is load-balanced across all of the links in the aggregated interface. If one or more links in the aggregated interface fails, the traffic is switched to the remaining links.

You can apply a Layer 2 port-mirroring *firewall filter* to an aggregated Ethernet interface to configure *port mirroring* at the parent interface. However, if any child interfaces are bound to different Layer 2 port-mirroring instances, packets received at the child interfaces will be mirrored to the destinations specified by their respective port-mirroring instances. Thus, multiple child interfaces can mirror packets to multiple destinations.

For example, suppose the parent aggregated Ethernet interface instance `ae0` has two child interfaces:

- `xe-2/0/0`
- `xe-3/1/2`

Suppose that these child interfaces on `ae0` are bound to two different Layer 2 port-mirroring instances:

- `pm_instance_A`—A named instance of Layer 2 port-mirroring, bound to child interface `xe-2/0/0`.
- `pm_instance_B`—A named instance of Layer 2 port-mirroring, bound to child interface `xe-3/1/2`.

Now suppose you apply a Layer 2 port-mirroring firewall filter to the Layer 2 traffic sent on `ae0.0` (logical unit `0` on the aggregated Ethernet interface instance `0`). This enables *port mirroring* on `ae0.0`, which has the following effect on the processing of traffic received on the child interfaces for which Layer 2 port-mirroring properties are specified:

- The packets received on `xe-2/0/0` are mirrored to the output interfaces configured in port-mirroring instance `pm_instance_A`.
- The packets received on `xe-3/1/2.0` are mirrored to the output interfaces configured in port-mirroring instance `pm_instance_B`.

Because `pm_instance_A` and `pm_instance_B` can specify different packet-selection properties or mirror destination properties, the packets received on `xe-2/0/0` and `xe-3/1/2.0` can mirror different packets to different destinations.

Applying Layer 2 Port Mirroring to a Logical Interface

You can apply a Layer 2 port-mirroring firewall filter to the input or to the output of a logical interface, including an aggregated Ethernet logical interface. Only packets of the address-type family specified by the filter action are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the input to a logical interface or output to a logical interface. For details, see Defining a Layer 2 Port-Mirroring Firewall Filter.

> **NOTE**: This configuration task shows two Layer 2 port-mirroring firewall filters: one filter applied to the logical interface ingress traffic, and one filter applied to the logical interface egress traffic.

To apply a Layer 2 port-mirroring firewall filter to an input or output logical interface:

1. Configure the underlying physical interface for the logical interface.

   a. Enable configuration of the underlying physical interface:

```
[edit]
user@host# edit interfaces interface-name
```

> **NOTE**: A port-mirroring firewall filter can also be applied to an aggregated-Ethernet logical interface.

   b. For Gigabit Ethernet interfaces and aggregated Ethernet interfaces configured for VPLS, enable the reception and transmission of 802.1Q VLAN-tagged frames on the interface:

```
[edit interfaces interface-name]
user@host# set vlan-tagging
```

   c. For Ethernet interfaces that have IEEE 802.1Q VLAN tagging and bridging enabled and that must accept packets carrying TPID 0x8100 or a user-defined TPID, set the logical link-layer encapsulation type:

```
[edit interfaces interface-name]
user@host# set encapsulation extended-vlan-ethernet-switching
```

2. Configure the logical interface to which you want to apply a Layer 2 port-mirroring firewall filter.

   a.

Specify the logical unit number:

```
[edit interfaces interface-name]
user@host# edit unit logical-unit-number
```

b. For a Gigabit Ethernet or Aggregated Ethernet interface, bind an 802.1Q VLAN tag ID to the logical interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@host# set vlan-id number
```

3. Enable specification of an input or output filter to be applied to Layer 2 packets that are part of bridging domain, Layer 2 switching cross-connects, or virtual private LAN service (VPLS).

   - If the filter is to be evaluated when packets are received on the interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@host# set family family filter input pm-filter-name-a
```

   - If the filter is to be evaluated when packets are sent on the interface:

```
[edit interfaces interface-name unit logical-unit-number]
user@host# set family family filter output pm-filter-name-b
```

   The value of the *family* option can be `ethernet-switching`, `ccc`, or `vpls`.

   NOTE: If port-mirroring firewall filters are applied at both the input and output of a logical interface, two copies of each packet are mirrored. To prevent the router or switch from forwarding duplicate packets to the same destination, include the optional `mirror-once` statement at the `[edit forwarding-options]` hierarchy level.

4. Verify the minimum configuration for applying a named Layer 2 port mirroring firewall filter to a logical interface:

```
[edit interfaces interface-name unit logical-unit-number family family filter ... ]
user@host# top
[edit]
user@host# show interfaces
```

```
interfaces {
    interface-name {
        vlan-tagging;
        encapsulation extended-vlan-ethernet-switching;
        unit number { # Apply a filter to the input of this interface
            vlan-id number;
            family (ethernet-switching | ccc | vpls) {
                filter {
                    input pm-filter-for-logical-interface-input;
                }
            }
        }
        unit number { # Apply a filter to the output of this interface
            vlan-id number;
            family (ethernet-switching | ccc | vpls) {
                filter {
                    output pm-filter-for-logical-interface-output;
                }
            }
        }
    }
}
```

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a Bridge Domain

You can apply a Layer 2 port-mirroring firewall filter to traffic being forwarded or flooded to a bridge domain. Only packets of the specified family type and forwarded or flooded to that bridge domain are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the traffic being forwarded to a bridge domain or flooded to a bridge domain. For details, see Defining a Layer 2 Port-Mirroring Firewall Filter.

> **NOTE**: This configuration task shows two Layer_2 port-mirroring firewall filters: one filter applied to the bridge domain forwarding table ingress traffic, and one filter applied to the bridge domain flood table ingress traffic.

To apply a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of a bridge domain:

1. Enable configuration of the bridge domain *bridge-domain-name* to which you want to apply a Layer 2 port-mirroring firewall filter for forwarded or flooded traffic:

   - For a bridge domain:

     ```
     [edit]
     user@host# edit bridge-domains bridge-domain-name
     ```

   - For a bridge domain under a routing instance:

     ```
     [edit]
     user@host# edit routing-instances routing-instance-name bridge-domains bridge-domain-name
     user@host# set instance-type virtual-switch
     ```

     For more detailed configuration information, see *Configuring a VPLS Routing Instance*.

2. Configure the bridge domain:

   ```
   [edit]
   user@host# set domain-type bridge
   user@host# set interface interface-name
   user@host# set routing-interface routing-interface-name
   ```

   For detailed configuration information, see Configuring a Bridge Domain and Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances.

3. Enable configuration of traffic forwarding on the bridge domain:

   ```
   [edit ... bridge-domains bridge-domain-name]
   user@host# edit forwarding-options
   ```

4. Apply a Layer 2 port-mirroring firewall filter to the bridge domain forwarding table or flood table.

   - To mirror packets being forwarded to the bridge domain:

     ```
     [edit ... bridge-domains bridge-domain-name forwarding-options]
     user@host# set filter input pm-filter-for-bd-ingress-forwarded
     ```

- To mirror packets being flooded to the bridge domain:

```
[edit ... bridge-domains bridge-domain-name forwarding-options]
user@host# set flood input pm-filter-for-bd-ingress-flooded
```

5. Verify the minimum configuration for applying a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of the bridge domain.

   a. Navigate to the hierarchy level at which the bridge domain is configured:

   - [edit]
   - [edit routing-instances *routing-instance-name*]

   b. Display the bridge domain configurations:

```
user@host# show bridge domains

bridge-domains {
    bridge-domain-name {
        instance-type virtual-switch; # For a bridge domain under a routing instance.
        domain-type bridge;
        interface interface-name;
        forwarding-options {
            filter { # Mirror ingress forwarded traffic
                input pm-filter-for-bd-ingress-forwarded;
            }
            flood { # Mirror ingress flooded traffic
                input pm-filter-for-bd-ingress-flooded;
            }
        }
    }
}
```

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VPLS Routing Instance

You can apply a Layer 2 port-mirroring firewall filter to traffic being forwarded or flooded to a VPLS routing instance. Only packets of the specified family type and forwarded or flooded to that VPLS routing instance are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the traffic being forwarded to a VPLS routing instance or flooded to a VLAN. For details, see Defining a Layer 2 Port-Mirroring Firewall Filter.

  NOTE: This configuration task shows two Layer_2 port-mirroring firewall filters: one filter applied to the VPLS routing instance forwarding table ingress traffic, and one filter applied to the VPLS routing instance flood table ingress traffic.

To apply a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of a VPLS routing instance:

1. Enable configuration of the VPLS routing instance to which you want to apply a Layer 2 port-mirroring firewall filter for forwarded or flooded traffic:

   ```
   [edit]
   user@host# edit routing-instances routing-instance-name
   user@host# set instance-type vpls
   user@host# set interface interface-name
   user@host# set route-distinguisher (as-number:number | ip-address:number)
   user@host# set vrf-import [policy-names]
   user@host# set vrf-export [policy-names]
   user@host# edit protocols vpls
   user@host@ ... vpls-configuration ...
   ```

   For more detailed configuration information, see *Configuring a VPLS Routing Instance*.

2. Enable configuration of traffic forwarding on the VPLS routing instance:

   ```
   [edit routing-instances routing-instance-name protocols vpls]
   user@host# up 2
   [edit routing-instances routing-instance-name]
   user@host# edit forwarding-options
   ```

3. Apply a Layer 2 port-mirroring firewall filter to the VPLS routing instance forwarding table or flood table.

   - To mirror packets being forwarded to the VPLS routing instance:

     ```
     [edit routing-instances routing-instance-name forwarding-options]
     user@host# set filter input pm-filter-for-vpls-ri-forwarded
     ```

- To mirror packets being flooded to the VPLS routing instance:

```
[edit routing-instances routing-instance-name forwarding-options]
user@host# set flood input pm-filter-for-vpls-ri-flooded
```

4. Verify the minimum configuration for applying a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of the VPLS routing instance:

```
[edit routing-instances routing-instance-name forwarding-options]
user@host# top
[edit]
user@host# show routing-instances

routing-instances {
    routing-instance-name {
        instance-type vpls;
        interface interface-name;
        route-distinguisher (as-number:number | ip-address:number);
        vrf-import [policy-names];
        vrf-export [policy-names];
        protocols {
            vpls {
                ...vpls-configuration...
            }
        }
        forwarding-options {
            family vpls {
                filter { # Mirror ingress forwarded traffic
                    input pm-filter-for-vpls-ri-forwarded;
                }
                flood { # Mirror ingress flooded traffic
                    input pm-filter-for-vpls-ri-flooded;
                }
            }
        }
    }
}
```

Applying Layer 2 Port Mirroring to Traffic Forwarded or Flooded to a VLAN

You can apply a Layer 2 port-mirroring firewall filter to traffic being forwarded or flooded to a VLAN. Only packets of the specified family type and forwarded or flooded to that VLAN are mirrored.

Before you begin, complete the following task:

- Define a Layer 2 port-mirroring firewall filter to be applied to the traffic being forwarded to a VLAN or flooded to a VLAN. For details, see Defining a Layer 2 Port-Mirroring Firewall Filter.

> **NOTE**: This configuration task shows two Layer_2 port-mirroring firewall filters: one filter applied to the VLAN forwarding table ingress traffic, and one filter applied to the VLAN flood table ingress traffic.

To apply a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of a VLAN:

1. Enable configuration of the VLAN ***bridge-domain-name*** to which you want to apply a Layer 2 port-mirroring firewall filter for forwarded or flooded traffic:
   - For a VLAN:

     ```
     [edit]
     user@host# edit bridge-domains bridge-domain-name
     ```

   - For a VLAN under a routing instance:

     ```
     [edit]
     user@host# edit routing-instances routing-instance-name bridge-domains bridge-domain-name
     user@host# set instance-type virtual-switch
     ```

     For more detailed configuration information, see *Configuring a VPLS Routing Instance*.
2. Configure the VLAN:

   ```
   [edit]
   user@host# set domain-type bridge
   user@host# set interface interface-name
   user@host# set routing-interface routing-interface-name
   ```

   For more detailed configuration information, see Configuring a Bridge Domain and Configuring VLAN Identifiers for Bridge Domains and VPLS Routing Instances.

3. Enable configuration of traffic forwarding on the VLAN:

```
[edit ... bridge-domains bridge-domain-name]
user@host# edit forwarding-options
```

4. Apply a Layer 2 port-mirroring firewall filter to the VLAN forwarding table or flood table.

   - To mirror packets being forwarded to the VLAN:

```
[edit ... bridge-domains bridge-domain-name forwarding-options]
user@host# set filter input pm-filter-for-bd-ingress-forwarded
```

   - To mirror packets being flooded to the VLAN:

```
[edit ... bridge-domains bridge-domain-name forwarding-options]
user@host# set flood input pm-filter-for-bd-ingress-flooded
```

5. Verify the minimum configuration for applying a Layer 2 port-mirroring firewall filter to the forwarding table or flood table of the VLAN.

   a. Navigate to the hierarchy level at which the VLAN is configured:

      - [edit]
      - [edit routing-instances *routing-instance-name*]

   b. Display the VLAN configurations:

```
user@host# show vlans

vlans {
    vlan-name {
        instance-type virtual-switch; # For a bridge domain under a routing instance.
        domain-type bridge;
        interface interface-name;
        forwarding-options {
            filter { # Mirror ingress forwarded traffic
                input pm-filter-for-bd-ingress-forwarded;
            }
            flood { # Mirror ingress flooded traffic
                input pm-filter-for-bd-ingress-flooded;
            }
```

```
        }
    }
}
```

Example: Layer 2 Port Mirroring at a Logical Interface

The following steps describe an example in which the global port-mirroring instance and a port-mirroring firewall filter are used to configure Layer 2 port mirroring for the input to a logical interface.

1. Configure the VLAN **example-bd-with-analyzer**, which contains the external packet analyzer, and the VLAN **example-bd-with-traffic**, which contains the source and destination of the Layer 2 traffic being mirrored:

   ```
   [edit]
   bridge-domains {
       example-bd-with-analyzer { # Contains an external traffic analyzer
           vlan-id 1000;
           interface ge-2/0/0.0; # External analyzer
       }
       example-bd-with-traffic { # Contains traffic input and output interfaces
           vlan-id 1000;
           interface ge-2/0/6.0; # Traffic input port
           interface ge-3/0/1.2; # Traffic output port
       }
   }
   ```

   Assume that logical interface **ge-2/0/0.0** is associated with an external traffic analyzer that is to receive port-mirrored packets. Assume that logical interfaces **ge-2/0/6.0** and **ge-3/0/1.2** will be traffic input and output ports, respectively.

2. Configure Layer 2 port-mirroring for the global instance, with the port-mirroring destination being the VLAN interface associated with the external analyzer (logical interface **ge-2/0/0.0** on VLAN **example-bd-with-analyzer**). Be sure to enable the option that allows filters to be applied to this port-mirroring destination:

   ```
   [edit]
   forwarding-options {
       port-mirroring {
           input {
               rate 10;
               run-length 5;
           }
   ```

```
        family ethernet-switching {
            output {
                interface ge-2/0/0.0; # Mirror packets to the external analyzer
                no-filter-check; # Allow filters on the mirror destination interface
            }
        }
    }
}
```

The `input` statement at the `[edit forwarding-options port-mirroring]` hierarchy level specifies that sampling begins every tenth packet and that each of the first five packets selected are to be mirrored.

The `output` statement at the `[edit forwarding-options port-mirroring family ethernet-switching]` hierarchy level specifies the output mirror interface for Layer 2 packets in a bridging environment:

- Logical interface **ge-2/0/0.0**, which is associated with the external packet analyzer, is configured as the port-mirroring destination.
- The optional `no-filter-check` statement allows filters to be configured on this destination interface.

3. Configure the Layer 2 port-mirroring firewall filter **example-bridge-pm-filter**:

```
[edit]
firewall {
    family ethernet-switching {
        filter example-bridge-pm-filter {
            term example-filter-terms {
                then {
                    accept;
                    port-mirror;
                }
            }
        }
    }
}
```

When this firewall filter is applied to the input or output of a logical interface for traffic in a bridging environment, Layer 2 port mirroring is performed according to the input packet-sampling properties and mirror destination properties configured for the Layer 2 port mirroring global instance. Because this firewall filter is configured with the single, default filter action **accept**, all packets selected by the **input** properties (**rate** = **10** and **run-length** = **5**) match this filter.

4. Configure the logical interfaces:

```
[edit]
interfaces {
    ge-2/0/0 { # Define the interface to the external analyzer
        encapsulation ethernet-bridge;
        unit 0 {
            family ethernet-switching;
        }
    }
    ge-2/0/6 { # Define the traffic input port
        flexible-vlan-tagging;
        encapsulation extended-vlan-bridge;
        unit 0 {
            vlan-id 100;
            family ethernet-switching {
                filter {
                    input example-bridge-pm-filter; # Apply the port-mirroring firewall filter
                }
            }
        }
    }
    ge-3/0/1 { # Define the traffic output port
        flexible-vlan-tagging;
        encapsulation extended-vlan-bridge;
        unit 2 {
            vlan-tags outer 10 inner 20;
            family ethernet-switching;
        }
    }
}
```

Packets received at logical interface **ge-2/0/6.0** on VLAN **example-bd-with-traffic** are evaluated by the port-mirroring firewall filter **example-bridge-pm-filter**. The firewall filter acts on the input traffic according to the filter actions configured in the firewall filter itself plus the input packet-sampling properties and mirror destination properties configured in the global port-mirroring instance:

- All packets received at **ge-2/0/6.0** are forwarded to their (assumed) normal destination at logical interface **ge-3/0/1.2**.
- For every ten input packets, copies of the first five packets in that selection are forwarded to the external analyzer at logical interface **ge-0/0/0.0** in the other VLAN, **example-bd-with-analyzer**.

If you configure the port-mirroring firewall filter **example-bridge-pm-filter** to take the **discard** action instead of the **accept** action, all original packets are discarded while copies of the packets selected using the global port-mirroring **input** properties are sent to the external analyzer.

Example: Layer 2 Port Mirroring for a Layer 2 VPN

The following example is not a complete configuration, but shows all the steps needed to configure port mirroring on an L2VPN using **family ccc**.

1. Configure the VLAN **port-mirror-bd**, which contains the external packet analyzer:

```
[edit]
vlans {
    port-mirror-vlan { # Contains an external traffic analyzer
        interface ge-2/2/9.0; # External analyzer
    }
}
```

2. Configure the Layer 2 VPN CCC to connect logical interface **ge-2/0/1.0** and logical interface **ge-2/0/1.1**:

```
[edit]
protocols {
    mpls {
        interface all;
    }
    connections {
        interface-switch if_switch {
            interface ge-2/0/1.0;
            interface ge-2/0/1.1;
        }
    }
}
```

3. Configure Layer 2 port mirroring for the global instance, with the port-mirroring destination being the VLAN interface associated with the external analyzer (logical interface **ge-2/2/9.0** on VLAN **example-bd-with-analyzer**):

```
[edit]
forwarding-options {
    port-mirroring {
```

```
        input {
            rate 1;
            maximum-packet-length 200;
        }
        family ccc {
            output {
                interface ge-2/2/9.0; # Mirror packets to the external analyzer
            }
        }
        instance {
            inst1 {
                input {
                    rate 1;
                    maximum-packet-length 300;
                }
                family ccc {
                    output {
                        interface ge-2/2/9.0;
                    }
                {
            }
        }
    }
}
```

4. Define the Layer 2 port-mirroring firewall filter **pm_filter_ccc** for **family ccc**:

```
[edit]
firewall {
    family ccc {
        filter pm_filter_ccc {
            term pm {
                then port-mirror;
            }
        }
    }
}
```

5. Apply the port mirror instance to the chassis:

```
[edit]
chassis {
    fpc 2 {
        port-mirror-instance inst1;
    }
}
```

6. Configure interface **ge-2/2/9** for the VLANs, and configure interface **ge-2/0/1** for port mirroring with the **pm_filter_ccc** firewall filter:

```
[edit]
interfaces {
    ge-2/2/9 {
        encapsulation ethernet-bridge;
        unit 0 {
            family ethernet-switching;
        }
    }
    ge-2/0/1 {
        vlan-tagging;
        encapsulation extended-vlan-ccc;
        unit 0 {
            vlan-id 10;
            family ccc {
                filter {
                    input pm_filter_ccc;
                }
            }
        }
        unit 1 {
            vlan-id 20;
            family ccc {
                filter {
                    output pm_filter_ccc;
                }
            }
        }
```

```
    }
}
```

Example: Layer 2 Port Mirroring for a Layer 2 VPN with LAG Links

The following example is not a complete configuration, but shows all the steps needed to configure port mirroring on an L2VPN using **family ccc** and aggregated Ethernet links.

1. Configure the VLAN **port_mirror_bd**, which contains the external packet analyzer:

```
[edit]
vlans {
    port_mirror_vlan { # Contains an external traffic analyzer
        interface ge-2/2/8.0; # External analyzer
    }
}
```

2. Configure the Layer 2 VPN CCC to connect interface **ae0.0** and interface **ae0.1**:

```
[edit]
protocols {
    mpls {
        interface all;
    }
    connections {
        interface-switch if_switch {
            interface ae0.0;
            interface ae0.1;
        }
    }
}
```

3. Configure Layer 2 port mirroring for the global instance, with the port-mirroring destination being the VLAN interface associated with the external analyzer (logical interface **ge-2/2/9.0** on VLAN **example_bd_with_analyzer**):

```
[edit]
forwarding-options {
    port-mirroring {
        input {
```

```
            rate 1;
            maximum-packet-length 200;
        }
        family ccc {
            output {
                interface ge-2/2/8.0; # Mirror packets to the external analyzer
            }
        }
        instance {
            pm_instance_1 {
                input {
                    rate 1;
                    maximum-packet-length 300;
                }
                family ccc {
                    output {
                        interface ge-2/2/8.0;
                    }
                {
            }
        }
    }
}
```

4. Configure the firewall filter **pm_ccc** for **family ccc**:

```
[edit]
firewall {
    family ccc {
        filter pm_ccc {
            term pm {
                then port-mirror;
            }
        }
    }
}
```

5. Apply the aggregated Ethernet interfaces and port mirror instance to the chassis:

```
[edit]
chassis {
```

```
    aggregated-devices {
        ethernet {
            device-count 10;
        }
    }
    fpc 2 {
        port-mirror-instance pm_instance_1;
    }
}
```

6. Configure interfaces **ae0** and **ge-2/0/2** (for aggregated Ethernet) and **ge-2/2/8** (for port mirroring) with the **pm_ccc** filter:

```
[edit]
interfaces {
    ae0 {
        vlan-tagging;
        encapsulation extended-vlan-ccc;
        unit 0 {
            vlan-id 10;
            family ccc {
                filter {
                    input pm_ccc;
                }
            }
        }
        unit 1 {
            vlan-id 20;
            family ccc {
                filter {
                    output pm_ccc;
                }
            }
        }
    }
    ge-2/0/2 {
        gigether-options {
            802.3ad ae0;
        }
    }
    ge-2/2/8 {
        encapsulation ethernet-bridge;
```

```
        unit 0 {
            family ethernet-switching;
        }
    }
}
```

Release History Table

Release	Description
13.3R6	Starting with Junos OS Release 13.3R6, only MPC interfaces support `family any` to do port mirroring.

Configuring Port Mirroring for Multiple Destinations

IN THIS SECTION

- Understanding Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups | **1185**
- Defining a Next-Hop Group on MX Series Routers for Port Mirroring | **1186**
- Example: Configuring Multiple Port Mirroring with Next-Hop Groups on M, MX and T Series Routers | **1188**
- Example: Layer 2 Port Mirroring to Multiple Destinations | **1193**

Understanding Layer 2 Port Mirroring to Multiple Destinations Using Next-Hop Groups

On an MX Series router and on an EX Series switch, you can mirror traffic to multiple destinations by configuring next-hop groups in Layer 2 port-mirroring firewall filters applied to tunnel interfaces. The mirroring of packets to multiple destinations is also known as *multipacket port mirroring*,

NOTE: Junos OS Release 9.5 introduced support for Layer 2 *port mirroring* using next-hop groups on MX Series routers, but required installation of a Tunnel PIC. Beginning in Junos OS Release 9.6, Layer 2 port mirroring using next-hop groups on MX Series routers does not require Tunnel PICs.

On MX Series routers and on EX Series switches, you can define a *firewall filter* for mirroring packets to a next-hop group. The next-hop group can contain Layer 2 members, Layer 3 members, and subgroups

that are either unit list (mirroring packets to each interface) or load-balanced (mirroring packets to one of several interfaces). The MX Series router and the EX Series switch supports up to 30 next-hop groups. Each next-hop group supports up to 16 next-hop addresses. Each next-hop group must specify at least two addresses.

To enable port mirroring to the members of a next-hop group, you specify the next-hop group as the filter action of a firewall filter, and then you apply the firewall filter to logical tunnel interfaces (`lt-`) or virtual tunnel interfaces (`vt-`) on the MX Series router or on the EX Series switch.

NOTE: The use of subgroups for load-balancing mirrored traffic is not supported.

Defining a Next-Hop Group on MX Series Routers for Port Mirroring

Starting with release 14.2, on routers containing an Internet Processor II application-specific integrated circuit (ASIC) or T Series Internet Processor, you can send a copy of an IP version 4 (IPv4) or IP version 6 (IPv6) packet from the router to an external host address or a packet analyzer for analysis. This is known as *port mirroring*.

Port mirroring is different from traffic sampling. In traffic sampling, a sampling key based on the IPv4 header is sent to the Routing Engine. There, the key can be placed in a file, or cflowd packets based on the key can be sent to a cflowd server. In port mirroring, the entire packet is copied and sent out through a next-hop interface.

You can configure simultaneous use of sampling and port mirroring, and set an independent sampling rate and run-length for port-mirrored packets. However, if a packet is selected for both sampling and port mirroring, only one action can be performed, and port mirroring takes precedence. For example, if you configure an interface to sample every packet input to the interface and a filter also selects the packet to be port mirrored to another interface, only the port mirroring takes effect. All other packets not matching the explicit filter port-mirroring criteria continue to be sampled when forwarded to their final destination.

Next-hop groups allow you to include port mirroring on multiple interfaces.

On MX Series routers, you can mirror tunnel interface input traffic to multiple destinations. To this form of multipacket port mirroring, you specify two or more destinations in a next-hop group, define a firewall filter that references the next-hop group as the filter action, and then apply the filter to a logical tunnel interface `lt-`) or virtual tunnel interfaces (`vt-` on the MX Series router.

To define a next-hop group for a Layer 2 port-mirroring firewall filter action:

1. Enable the configuration of forwarding options.

```
[edit]
user@host set forwarding-options port-mirroring family (inet | inet6) output
```

2. Enable configuration of a next-hop-group for Layer 2 port mirroring.

```
[edit forwarding-options port-mirroring ... family (inet | inet6) output]
user@host# set next-hop-group next-hop-group-name
```

3. Specify the type of addresses to be used in the next-hop group configuration.

```
[edit forwarding-options port-mirroring ... family (inet | inet6) output next-hop-group next-hop-group-name]
user@host# set group-type inet6
```

4. Specify the interfaces of the next-hop route.

```
[edit forwarding-options port-mirroring ... family (inet | inet6) output next-hop-group next-hop-group-name]
user@host# set interface logical-interface-name-1
user@host# set interface logical-interface-name-2
```

or

```
[edit forwarding-options port-mirroring ... family (inet | inet6) output next-hop-group next-hop-group-name]
user@host# set interface interface-name next-hop next-hop-address
```

The MX Series router supports up to 30 next-hop groups. Each next-hop group supports up to 16 next-hop addresses. Each next-hop group must specify at least two addresses. The *next-hop-address* can be an IPv4 or IPv6 address.

5. (Optional) Specify the next-hop subgroup.

```
[edit forwarding-options port-mirroring ... family (inet | inet6) output next-hop-group next-hop-group-name]
user@host# set next-hop-subgroup subgroup-name interface interface-name next-hop next-hop-address
```

6. Verify the configuration of the next-hop group.

```
[edit forwarding-options port-mirroring ... family (inet | inet6) output next-hop-group next-hop-group-name]
user@host# top
[edit]
user@host# show forwarding-options

...
next-hop-group next-hop-group-name {
    group-type inet6;
    interface logical-interface-name-1;
    interface interface-name{
        next-hop next-hop-address;
    }
    next-hop-subgroup subgroup-name{
        interface interface-name{
            next-hop next-hop-address;
        }
    }
}
...
```

Example: Configuring Multiple Port Mirroring with Next-Hop Groups on M, MX and T Series Routers

When you need to analyze traffic containing more than one packet type, or you wish to perform multiple types of analysis on a single type of traffic, you can implement multiple port mirroring and next-hop groups. You can make up to 16 copies of traffic per group and send the traffic to next-hop group members. A maximum of 30 groups can be configured on a router at any given time. The port-mirrored traffic can be sent to any interface, except aggregated SONET/SDH, aggregated Ethernet, loopback (lo0), or administrative (**fxp0**) interfaces. To send port-mirrored traffic to multiple flow servers or packet analyzers, you can use the `next-hop-group` statement at the `[edit forwarding-options]` hierarchy level.

Figure 43: Active Flow Monitoring—Multiple Port Mirroring with Next-Hop Groups Topology Diagram

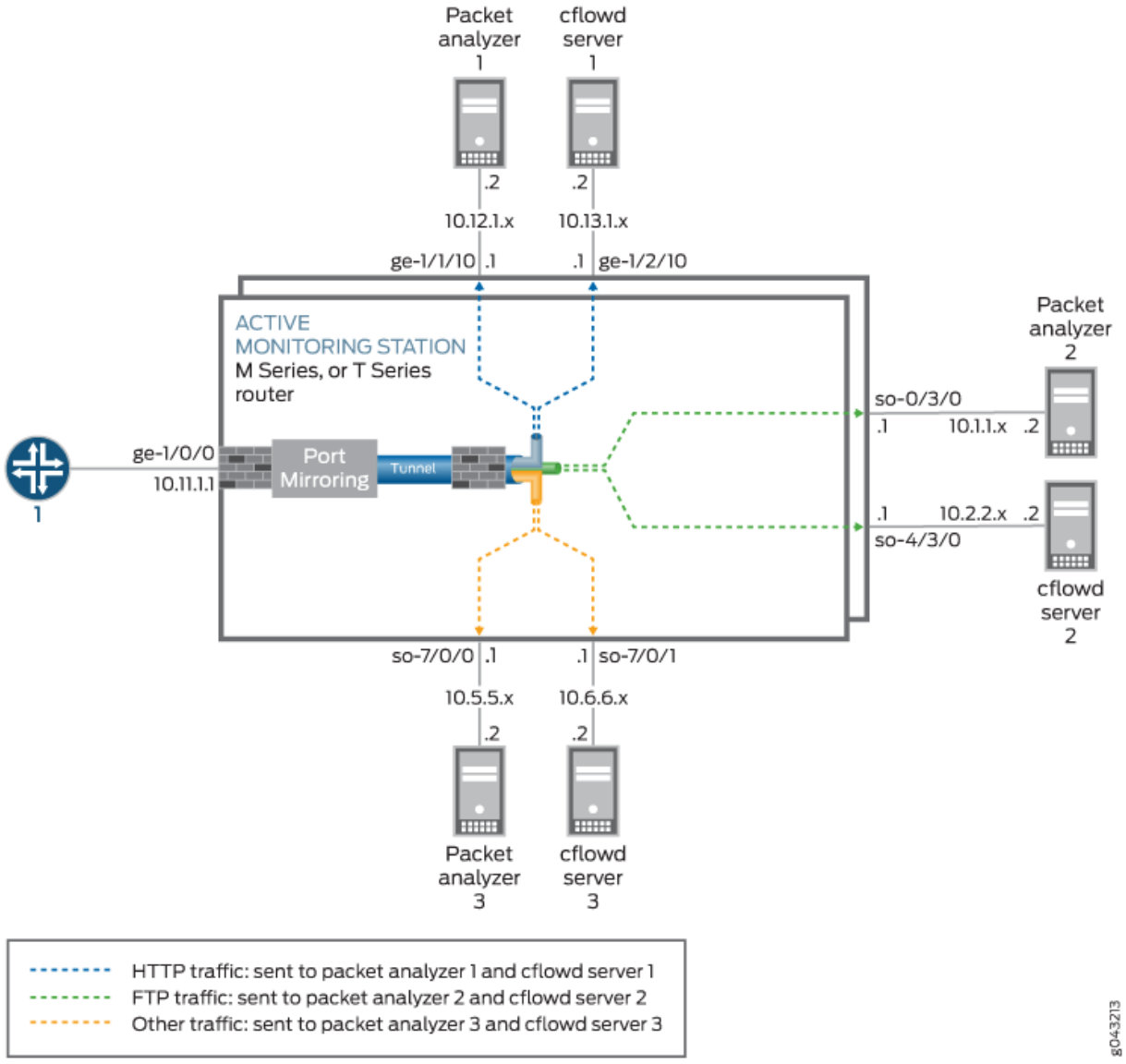


Figure 43 on page 1189 shows an example of how to configure multiple port mirroring with next-hop groups. All traffic enters the monitoring router at interface ge-1/0/0. A firewall filter counts and port-mirrors all incoming packets to a Tunnel Services PIC. A second filter is applied to the tunnel interface and splits the traffic into three categories: HTTP traffic, FTP traffic, and all other traffic. The three types of traffic are assigned to three separate next-hop groups. Each next-hop group contains a unique pair of exit interfaces that lead to different groups of packet analyzers and flow servers.

NOTE: Instances enabled to mirror packets to different destinations from the same PFE, also use different sampling parameters for each instance. When we configure Layer2 Port-mirroring with both global port-mirroring and instance based port-mirroring, PIC level instances will override FPC level and the FPC level will override the Global instance.

```
[edit]
interfaces {
                  ge-1/0/0 { # This is the input interface where packets enter the router.
        unit 0 {
            family inet {
                filter {
                                        input mirror_pkts; # Here is where you apply the first
filter.
                }
                address 10.11.1.1/24;
            }
        }
    }
                  ge-1/1/0 { # This is an exit interface for HTTP packets.
        unit 0 {
            family inet {
                address 10.12.1.1/24;
            }
        }
    }
                  ge-1/2/0 { # This is an exit interface for HTTP packets.
        unit 0 {
            family inet {
                address 10.13.1.1/24;
            }
        }
    }
                  so-0/3/0 { # This is an exit interface for FTP packets.
        unit 0 {
            family inet {
                address 10.1.1.1/30;
            }
        }
    }
```

```
                so-4/3/0 { # This is an exit interface for FTP packets.
        unit 0 {
            family inet {
                address 10.2.2.1/30;
            }
        }
    }
                so-7/0/0 { # This is an exit interface for all remaining packets.
        unit 0 {
            family inet {
                address 10.5.5.1/30;
            }
        }
    }
                so-7/0/1 { # This is an exit interface for all remaining packets.
        unit 0 {
            family inet {
                address 10.6.6.1/30;
            }
        }
    }
                vt-3/3/0 { # The tunnel interface is where you send the port-mirrored traffic.
        unit 0 {
            family inet;
        }
        unit 1 {
            family inet {
                filter {
                                        input collect_pkts;  # This is where you apply the
second firewall filter.
                }
            }
        }
    }
}
forwarding-options {
                port-mirroring { # This is required when you configure next-hop groups.
        family inet {
            input {
                                    rate 1; # This port-mirrors all packets (one copy for every
packet received).
            }
                                output { # Sends traffic to a tunnel interface to enable
```

```
multiport mirroring.
                interface vt-3/3/0.1;
                no-filter-check;
            }
        }
    }
    next-hop-group ftp-traffic { # Point-to-point interfaces require you to specify the
        interface so-4/3/0.0; # interface name.
        interface so-0/3/0.0;
    }
    next-hop-group http-traffic { # Configure a next hop for all multipoint interfaces.
        interface ge-1/1/0.0 {
            next-hop 10.12.1.2;
        }
        interface ge-1/2/0.0 {
            next-hop 10.13.1.2;
        }
    }
    next-hop-group default-collect {
        interface so-7/0/0.0;
        interface so-7/0/1.0;
    }
}
firewall {
    family inet {
                        filter mirror_pkts { # Apply this filter to the input interface.
            term catch_all {
                then {
                    count input_mirror_pkts;
                                        port-mirror; # This action sends traffic to be copied
and port-mirrored.
                }
            }
        }
                        filter collect_pkts { # Apply this filter to the tunnel interface.
                              term ftp-term { # This term sends FTP traffic to an FTP next-hop
group.
                from {
                    protocol ftp;
                }
                                    then next-hop-group ftp-traffic;
            }
                                    term http-term { # This term sends HTTP traffic to an HTTP next-
```

```
hop group.
                from {
                    protocol http;
                }
                                then next-hop-group http-traffic;
            }
                               term default { # This sends all remaining traffic to a final next-
hop group.
                                then next-hop-group default-collectors;
            }
        }
    }
}
```

Example: Layer 2 Port Mirroring to Multiple Destinations

On MX Series routers, you can mirror traffic to multiple destinations by configuring next-hop groups in Layer 2 port-mirroring firewall filters applied to tunnel interfaces.

1. Configure the chassis to support tunnel services at PIC 0 on FPC 2. This configuration includes two logical tunnel interfaces on FPC 2, PIC 0, port 10.

```
[edit]
chassis {
    fpc 2 {
        pic 0 {
            tunnel-services {
                bandwidth 1g;
            }
        }
    }
}
```

2. Configure the physical and logical interfaces for three bridge domains and one Layer 2 VPN CCC:

   - Bridge domain **bd** will span logical interfaces **ge-2/0/1.0** and **ge-2/0/1.1**.
   - Bridge domain **bd_next_hop_group** will span logical interfaces **ge-2/2/9.0** and **ge-2/0/2.0**.
   - Bridge domain **bd_port_mirror** will use the logical tunnel interface **lt-2/0/10.2**.

- Layer 2 VPN CCC **if_switch** will connect logical interfaces **ge-2/0/1.2** and **lt-2/0/10.1**.

```
[edit]
interfaces {
    ge-2/0/1 {
        flexible-vlan-tagging;
        encapsulation flexible-ethernet-services;
        unit 0 { # An interface on bridge domain 'bd'.
            encapsulation vlan-bridge;
            vlan-id 200;
            family bridge {
                filter {
                    input pm_bridge;
                }
            }
        }
        unit 1 { # An interface on bridge domain 'bd'.
            encapsulation vlan-bridge;
            vlan-id 201;
            family bridge {
                filter {
                    input pm_bridge;
                }
            }
        }
        unit 2 {
            encapsulation vlan-ccc;
            vlan-id 1000;
        }
    }
    ge-2/0/2 { # For 'bd_next_hop_group'
        encapsulation ethernet-bridge;
        unit 0 {
            family bridge;
        }
    }
    lt-2/0/10 {
        unit 1 {
            encapsulation ethernet-ccc;
            peer-unit 2;
        }
        unit 2 {
```

```
            encapsulation ethernet-bridge;
            peer-unit 1;
            family bridge {
                filter {
                    output redirect_to_nhg;
                }
            }
        }
    }
    ge-2/2/9 {
        encapsulation ethernet-bridge;
        unit 0 { # For 'bd_next_hop_group'
            family bridge;
        }
    }
}
```

3. Configure the three bridge domains and the Layer 2 VPN switching CCC:

   - Bridge domain **bd** spans logical interfaces **ge-2/0/1.0** and **ge-2/0/1.1**.
   - Bridge domain **bd_next_hop_group** spans logical interfaces **ge-2/2/9.0** and **ge-2/0/2.0**.
   - Bridge domain **bd_port_mirror** uses the logical tunnel interface **lt-2/0/10.2**.
   - Layer 2 VPN CCC **if_switch** connects interfaces **ge-2/0/1.2** and **lt-2/0/10.1**.

```
[edit]
bridge-domains {
    bd {
        interface ge-2/0/1.0;
        interface ge-2/0/1.1;
    }
    bd_next_hop_group {
        interface ge-2/2/9.0;
        interface ge-2/0/2.0;
    }
    bd_port_mirror {
        interface lt-2/0/10.2;
    }
}
protocols {
    mpls {
```

```
        interface all;
    }
    connections {
        interface-switch if_switch {
            interface ge-2/0/1.2;
            interface lt-2/0/10.1;
        }
    }
}
```

For detailed information about configuring the CCC connection for Layer 2 switching cross-connects, see the MPLS Applications User Guide.

4. Configure forwarding options:

   - Configure global port-mirroring properties to mirror **family vpls** traffic to an interface on the bridge domain **bd_port_mirror**.
   - Configure the next-hop group **nhg_mirror_to_bd** to forward Layer 2 traffic to the bridge domain **bd_next_hop_group**.

   Both of these forwarding options will be referenced by the port-mirroring firewall filter:

```
[edit]
forwarding-options {
    port-mirroring { # Global port mirroring properties.
        input {
            rate 1;
        }
        family vpls {
            output {
                interface lt-2/0/10.2; # Interface on 'bd_port_mirror' bridge domain.
                no-filter-check;
            }
        }
    }
    next-hop-group nhg_mirror_to_bd { # Configure a next-hop group.
        group-type layer-2; # Specify 'layer-2' for Layer 2; default 'inet' is for Layer 3.
        interface ge-2/0/2.0; # Interface on 'bd_next_hop_group' bridge domain.
        interface ge-2/2/9.0; # Interface on 'bd_next_hop_group' bridge domain.
    }
}
```

5. Configure two Layer 2 port-mirroring firewall filters for **family bridge** traffic:

   - **filter_pm_bridge**—Sends all **family bridge** traffic to the global port mirroring destination.
   - **filter_redirect_to_nhg**—Sends all **family bridge** traffic to the final next-hop group **nhg_mirror_to_bd**.

   Layer 2 port-mirroring firewall filters for **family bridge** traffic applies to traffic on a physical interface configured with encapsulation **ethernet-bridge**.

```
[edit]
firewall {
    family bridge {
        filter filter_pm_bridge {
            term term_port_mirror {
                then port-mirror;
            }
        }
        filter filter_redirect_to_nhg {
            term term_nhg {
                then next-hop-group nhg_mirror_to_bd;
            }
        }
    }
}
```

Release History Table

Release	Description
14.2	Starting with release 14.2, on routers containing an Internet Processor II application-specific integrated circuit (ASIC) or T Series Internet Processor, you can send a copy of an IP version 4 (IPv4) or IP version 6 (IPv6) packet from the router to an external host address or a packet analyzer for analysis.

Configuring Port Mirroring for Remote Destinations

IN THIS SECTION

Layer 2 Port Mirroring to Remote Destination by Using Destination as VLAN | **1198**

Configuration Layer 2 Port Mirroring to a Remote VLAN | 1198
Example: Configuring Layer 2 Port Mirroring to Remote VLAN | 1201

Layer 2 Port Mirroring to Remote Destination by Using Destination as VLAN

You configure port mirroring on an EX9200 switch to send copies of traffic to an output destination, such as an interface, a routing-instance, or a VLAN; and for the input traffic, you can configure a firewall filter term with various match conditions and actions.

When you configure VLAN as the output destination in a port-mirroring configuration, the traffic for each port-mirroring session is carried over a user-specified VLAN that is dedicated for that mirroring session in all participating switches. The mirrored traffic is copied onto that VLAN (also called as mirror VLAN) and forwarded to interfaces, which are members of the mirror VLAN. The destination interfaces, which are members of the mirror VLAN, can span across multiple switches in the network provided that the same remote mirroring VLAN is used for a mirroring session in all the switches.

You can use the `port-mirror` or `port-mirror-instance` action in the firewall filter configuration when you mirror traffic to remote destinations by configuring a VLAN as a port-mirroring output destination.

Configuration Layer 2 Port Mirroring to a Remote VLAN

IN THIS SECTION

Configuring Port Mirroring to a Remote VLAN | 1199

EX9200 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy the following packets:

- Packets entering or exiting a port
- Packets entering or exiting a VLAN

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable port mirroring that you have configured when you are not using them.
- Specify individual interfaces as input rather than specifying all interfaces as input in a port mirroring configuration.
- Limit the amount of mirrored traffic by:
  - Using statistical sampling.
  - Setting ratios to select statistical samples.
  - Using firewall filters.

Configuring Port Mirroring to a Remote VLAN

To filter packets to be mirrored to a port-mirroring instance, create the instance and then use it as the action in the firewall filter. You can use firewall filters in both local and remote mirroring configurations.

If the same port-mirroring instance is used in multiple filters or terms, the packets are copied to the port-mirroring output port or port-mirroring VLAN only once.

To filter mirrored traffic, create a port-mirroring instance under the `[edit forwarding-options]` hierarchy level, and then create a firewall filter. The filter can use any of the available match conditions and must have `port-mirror-instance` *instance-name* as an action. This action in the firewall filter configuration provides the input to the port-mirroring instance.

To configure a port-mirroring instance with firewall filters:

1. Configure the port-mirroring instance name and set the output destination to a VLAN:

```
[edit forwarding-options]
user@switch# set port-mirroring instance instance-name output vlan (vlan-ID | vlan-name)
```

   For example, configure a port-mirroring instance `employee-monitor` and set the output destination to a VLAN ID `999`:

```
[edit forwarding-options]
user@switch# set port-mirroring instance employee-monitor output vlan 999
```

2. Create a firewall filter by using any of the available match conditions and assign the port-mirroring instance name as an action in the firewall filter configuration.

```
[edit firewall family ethernet-switching]
user@switch set filter filter-name term term-name from match-condition
user@switch set filter filter-name term term-name then match-condition
user@switch# set filter filter-name term term-namethen port-mirror-instance instance-name
```

For example, create a firewall filter called `example-filter` with two terms `no-analyzer` and `to-analyzer`, and assign the `to-analyzer` term to the `employee-monitor` port-mirroring instance:

a. Create the first term to define the traffic that should not pass through to the port-mirroring instance `employee-monitor`:

```
[edit firewall family ethernet-switching]
user@switch# set filter (Firewall Filters) example-filter term no-analyzer from source-
address 192.0.2.14
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term no-analyzer from protocol tcp
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term no-analyzer then accept
```

b. Create the second term to define the traffic that should pass through to the port-mirroring instance `employee-monitor`:

```
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer from destination-port 80
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer then port-mirror-instance employee-
monitor
[edit firewall family ethernet-switching]
user@switch# set filter example-filter term to-analyzer then accept
```

3. Apply the firewall filter to an interface or VLAN that provides input to the port-mirroring instance.

To apply a firewall filter to an interface:

```
[edit]
user@switch# set interfaces interface-name unit 0 family ethernet-switching filer (input |
output) filter-name
```

To apply a firewall filter to a VLAN:

```
[edit]
user@switch# set vlan (vlan-ID or vlan-name) filter (input | output) filter-name
```

For example, to apply the `example-filter` firewall filter to the ge-0/0/1 interface:

```
[edit]
user@switch# set interfaces ge-0/0/1 unit 0 family ethernet-switching filter input example-
filter
```

For example, to apply the `example-filter` filter to the `source-vlan` VLAN:

```
[edit]
user@switch# set vlan source-vlan filter input example-filter
```

Example: Configuring Layer 2 Port Mirroring to Remote VLAN

IN THIS SECTION

- Requirements | **1202**
- Overview and Topology | **1202**
- Mirroring Employee-to-Web Traffic for Remote Analysis | **1203**
- Verification | **1209**

EX9200 switches enable you to configure mirroring to send copies of packets to either a local interface for local monitoring or to a VLAN for remote monitoring. You can use mirroring to copy these packets:

- Packets entering or exiting a port
- Packets entering or existing a VLAN

You can analyze the mirrored traffic by using a protocol analyzer application running on a remote monitoring station if you are sending mirrored traffic to an analyzer VLAN.

This topic includes two related examples that describe how to mirror traffic entering ports on the switch to the `remote-analyzer` VLAN so that you can perform analysis from a remote monitoring station. The first example shows how to mirror all traffic entering the ports connected to employee computers. The

second example shows the same scenario but includes a filter to mirror only the employee traffic going to the Web.

BEST PRACTICE: Mirror only necessary packets to reduce potential performance impact. We recommend that you:

- Disable your configured mirroring sessions when you are not using them.
- Specify individual interfaces as input to analyzers rather than specifying all interfaces as input.
- Limit the amount of mirrored traffic by using firewall filters.

This example describes how to configure remote mirroring:

Requirements

Before you configure remote mirroring, be sure that:

- You have an understanding of mirroring concepts.
- The interfaces that port-mirroring will use as output interfaces have been configured on the switch.

Overview and Topology

IN THIS SECTION

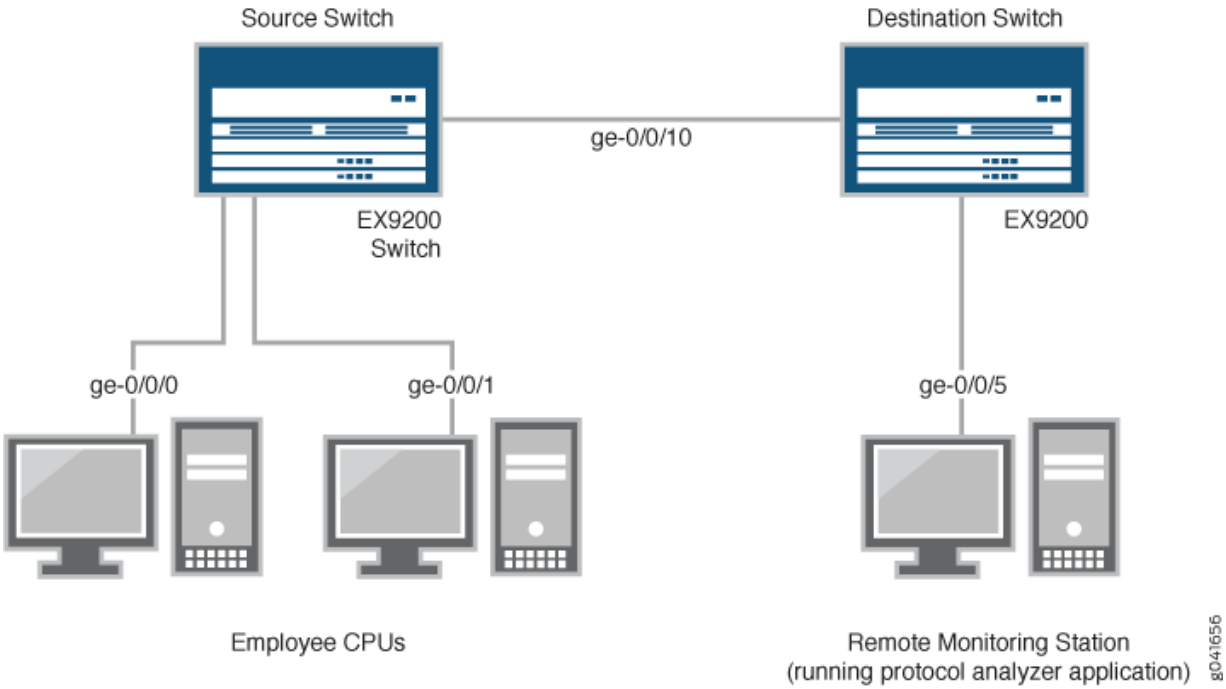
Topology | **1203**

This topic includes two related examples that describe how to configure mirroring to the `remote-analyzer` VLAN so that analysis can be performed from a remote monitoring station. The first example shows how to configure a switch to mirror all traffic from employee computers. The second example shows the same scenario, but the setup includes a filter to mirror only the employee traffic going to the Web.

Figure 44 on page 1203 shows the network topology for both these example scenarios.

Topology

Figure 44: Remote Mirroring Network Topology Example



In this example:

1. Interface ge-0/0/0 is a Layer 2 interface, and interface ge-0/0/1 is a Layer 2 interface (both interfaces on the source switch) that serve as connections for employee computers.
2. Interface ge-0/0/10 is a Layer 2 interface that connects the source switch to the destination switch.
3. Interface ge-0/0/5 is a Layer 2 interface that connects the destination switch to the remote monitoring station.
4. VLAN `remote-analyzer` is configured on all switches in the topology to carry the mirrored traffic.

Mirroring Employee-to-Web Traffic for Remote Analysis

IN THIS SECTION

- Procedure | 1204

To configure port mirroring for remote traffic analysis of employee-to-Web traffic, perform these tasks:

Procedure

CLI Quick Configuration

To quickly configure port-mirroring to mirror employee traffic to the external Web, copy the following commands and paste them into the switch terminal window:

- Copy and paste the following commands in the source switch terminal window:

```
[edit]
set forwarding-options port-mirroring instance employee-web-monitor output vlan 999
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode access
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
 set firewall family ethernet-switching filter watch-employee term employee-to-corp from
destination-address 192.0.2.16/28
 set firewall family ethernet-switching filter watch-employee term employee-to-corp from
source-address 192.0.2.16/28
 set firewall family ethernet-switching filter watch-employee term employee-to-corp then
accept
 set firewall family ethernet-switching filter watch-employee term employee-to-web from
destination-port 80
 set firewall family ethernet-switching filter watch-employee term employee-to-web then port-
mirror-instance employee-web-monitor
set interfaces ge-0/0/0 unit 0 family ethernet-switching filter input watch-employee
 set interfaces ge-0/0/1 unit 0 family ethernet-switching filter input watch-employee
```

- Copy and paste the following commands in the destination switch terminal window:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode access
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
 set interfaces ge-0/0/5 unit 0 family ethernet-switching interface-mode access
set interfaces ge-0/0/5 unit 0 family ethernet-switching vlan members 999
```

Step-by-Step Procedure

To configure port mirroring of all traffic from the two ports connected to employee computers to the `remote-analyzer` VLAN for use from a remote monitoring station:

1. On the source switch:

   a. Configure the `employee-web-monitor` port-mirroring instance:

   ```
   [edit ]
   user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching port mode access
   user@switch# set forwarding-options port-mirroring instance employee-web-monitor output
   vlan 999
   ```

   b. Configure the VLAN ID for the `remote-analyzer` VLAN:

   ```
   [edit vlans]
   user@switch# set remote-analyzer vlan-id 999
   ```

   c. Configure the interface to associate it with the `remote-analyzer` VLAN:

   ```
   [edit interfaces]
   user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
   ```

   d. Configure the firewall filter called `watch-employee`:

   ```
   [edit firewall family ethernet-switching]
   user@switch# set filter watch-employee term employee-to-corp from destination-address
   192.0.2.16/28
   user@switch# set filter watch-employee term employee-to-corp from source-address
   192.0.2.16/28
   user@switch# set filter watch-employee term employee-to-corp then accept
   user@switch# set filter watch-employee term employee-to-web from destination-port 80
   user@switch# set filter watch-employee term employee-to-web then port-mirror-instance
   employee-web-monitor
   ```

   In this configuration, the `employee-to-corp` term defines that traffic from destination-address `192.0.2.16/28` and source address `192.0.2.16/28` can be accepted to pass through the switch, and the `employee-to-web` term defines that traffic from port `80` must be sent to the port-mirroring instance `employee-web-monitor`.

e. Apply the firewall filter to the employee interfaces:

```
[edit interfaces]
user@switch# set ge-0/0/0 unit 0 family ethernet-switching filter input watch-employee
user@switch# set ge-0/0/1 unit 0 family ethernet-switching filter input watch-employee
```

2. On the destination switch:

- Configure the VLAN ID for the `remote-analyzer` VLAN:

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

- Configure the interface on the destination switch for access mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/10 unit 0 family ethernet-switching interface-mode access
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

- Configure the interface connected to the destination switch for access mode and associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set ge-0/0/5 unit 0 family ethernet-switching interface-mode access
user@switch# set ge-0/0/5 unit 0 family ethernet-switching vlan members 999
```

Results

Check the results of the configuration on the source switch:

```
[edit]
user@switch> show
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
```

```
                vlan {
                    members remote-analyzer;
                }
            }
        }
    }
    ge-0/0/0 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
}
firewall {
     family ethernet-switching {
        filter watch-employee {
            term employee-to-corp {
                from {
                    source-address {
                        192.0.2.16/28;
                    }
                    destination-address {
                        192.0.2.16/28;
                    }
                }
                then accept;
            }
            term employee-to-web {
                from {
                    destination-port 80;
                }
```

```
                then port-mirror-instance employee-web-monitor;
            }
        }
    }
}
 forwarding-options {
    analyzer employee-web-monitor {
        output {
            vlan {
                999;
            }
        }
    }
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
```

Check the results of the configuration on the destination switch:

```
[edit]
user@switch> show
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
interfaces {
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode access;
                vlan {
                    members remote-analyzer;
                }
            }
        }
    }
    ge-0/0/5 {
        unit 0 {
            family ethernet-switching {
```

```
                interface-mode access;
                vlan {
                    members remote-analyzer;
                }
            }
        }
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Port-Mirroring Instance Has Been Correctly Created | **1209**

To confirm that the configuration is working properly, perform these tasks:

Verifying That the Port-Mirroring Instance Has Been Correctly Created

Purpose

Verify that the port-mirror instance `employee-web-monitor` has been created on the switch with the appropriate output VLAN.

Action

You can verify that the port-mirror is configured as expected by using the `show forwarding-options port-mirror` command. To view previously created analyzers that are disabled, go to the J-Web interface.

To verify that the port-mirror is configured as expected while monitoring employee traffic on the source switch, run the `show forwarding-options port-miror` command on the source switch. The following output is displayed for this configuration example:

```
user@switch> show forwarding-options port-mirror

Instance Name: employee-web-monitor
  Instance Id: 3
  Input parameters:
```

```
    Rate                 : 1
    Run-length           : 0
    Maximum-packet-length : 0
  Output parameters:
    Family              State     Destination                          Next-hop
    ethernet-switching  up       default-switch/remote-analyzer
```

Meaning

This output shows that the `employee-web-monitor` instance has a ratio of 1 (mirroring every packet, which is the default), the maximum size of the original packet that was mirrored (0 indicates the entire packet), the state of the configuration is up (which indicates the proper state and that the analyzer is programmed, is mirroring the traffic entering ge-0/0/0 and ge-0/0/1, and is sending the mirrored traffic to the VLAN called `remote-analyzer`).

Configuring Port Mirroring Local and Remote Analysis

IN THIS SECTION

- Configuring Port Mirroring | **1210**
- Configuring Port Mirroring on SRX Series Firewalls | **1214**
- Examples: Configuring Port Mirroring for Local Analysis | **1217**
- Example: Mirroring Employee Web Traffic with a Firewall Filter | **1221**
- Example: Configuring Port Mirroring for Remote Analysis | **1226**

Configuring Port Mirroring

IN THIS SECTION

- Configuring Port Mirroring for Local Analysis | **1211**
- Configuring Port Mirroring for Remote Analysis | **1212**
- Filtering the Traffic Entering an Analyzer | **1213**

You use port mirroring to copy packets and send the copies to a device running an application such as a network analyzer or intrusion detection application so that you can analyze traffic without delaying it. You can mirror traffic entering or exiting a port or entering a VLAN, and you can send the copies to a local access interface or to a VLAN through a trunk interface.

We recommend that you disable port mirroring when you are not using it. To avoid creating a performance issue If you do enable port mirroring, we recommend that you select specific input interfaces instead of using the `all` keyword. You can also limit the amount of mirrored traffic by using a firewall filter.

NOTE: This task uses the Enhanced Layer 2 Software (ELS) configuration style. If your switch runs software that does not support ELS, see Configuring Port Mirroring. For ELS details, see Using the Enhanced Layer 2 Software CLI.

NOTE: If you want to create additional analyzers without deleting an existing analyzer, first disable the existing analyzer by using the **disable analyzer *analyzer-name*** command.

NOTE: You must configure port mirroring output interfaces as **family ethernet-switching**.

Configuring Port Mirroring for Local Analysis

To mirror interface traffic to a local interface on the switch:

1. If you want to mirror traffic that is ingressing or egressing specific interfaces, choose a name for the port-mirroring configuration and configure what traffic should be mirrored by specifying the interfaces and direction of traffic:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input (ingress | egress) interface interface-name
```

NOTE: If you configure Junos OS to mirror egress packets, do not configure more than 2000 VLANs. If you do so, some VLAN packets might contain incorrect VLAN IDs.

NOTE: If you configure mirroring for packets that egress an access interface, the original packets lose any VLAN tags when they exit the access interface, but the mirrored (copied) packets retain the VLAN tags when they are sent to the analyzer system.

2. If you want to specify that all traffic entering a VLAN should be mirrored, choose a name for the port-mirroring configuration and specify the VLAN:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input ingress vlan vlan-name
```

NOTE: You cannot configure port mirroring to copy traffic that egresses a VLAN.

3. Configure the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output interface interface-name
```

Configuring Port Mirroring for Remote Analysis

To mirror traffic to a VLAN for analysis at a remote location:

1. Configure a VLAN to carry the mirrored traffic:

```
[edit]
user@switch# set vlans vlan-name vlan-id number
```

2. Configure the interface that connects to another switch (the uplink interface) to trunk mode and associate it with the appropriate VLAN:

```
[edit]
user@switch# set interfaces interface-name unit 0 family ethernet-switching port-mode trunk
vlan members (vlan-name | vlan-id)
```

3. Configure the analyzer:

a. Choose a name for the analyzer:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name
```

b. Specify the interface to be mirrored and whether the traffic should be mirrored on ingress or egress:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name input (ingress | egress) interface interface-name
```

c. Specify the appropriate IP address or VLAN as the output (a VLAN is specified in this example:

```
[edit forwarding-options]
user@switch# set analyzer analyzer-name output vlan (vlan-name | vlan-id)
```

If you specify an IP address as the output, note the following constraints:

- The address cannot be in the same subnetwork as any of the switch management interfaces.
- If you create virtual routing instances and also create an analyzer configuration that includes an output IP address, the output address belongs to the default virtual routing instance (`inet.0` routing table).
- The analyzer device must be able to de-encapsulate GRE-encapsulated packets, or the GRE-encapsulated packets must be de-encapsulated before reaching the analyzer device. (You can use a network sniffer to de-encapsulate the packets.)

Filtering the Traffic Entering an Analyzer

NOTE: This functionality is not supported on NFX150 devices.

In addition to specifying which traffic to mirror by configuring an analyzer, you can also use a firewall filter to exercise more control over which packets are copied. For example, you might use a filter to specify that only traffic from certain applications be mirrored. The filter can use any of the available match conditions and must have an action of modifier of `port-mirror-instance` *instance-name*. If you use the same analyzer in multiple filters or terms, the output packets are copied only once.

When you use a firewall filter as the input to a port-mirroring instance, you send the copied traffic to a local interface or a VLAN just as you do when a firewall is not involved.

To configure port mirroring with filters:

1. Configure a port-mirroring instance for local or remote analysis. Configure only the output. For example, for local analysis enter:

```
[edit forwarding-options]
user@switch# set port-mirroring-instance instance-name output interface interface-name
```

> **NOTE**: You cannot configure input to this instance.

2. Create a firewall filter using any of the available match conditions. In a `then` term, specify include the action modifier `port-mirror-instance` *`instance-name`*.
3. Apply the firewall filter to the interfaces or VLAN that should provide the input to the analyzer:

```
[edit]
user@switch# set interfaces interface-name unit 0 family ethernet-switching filter input
filter-name

[edit]
user@switch# set vlan (vlan-name | vlan-id) filter input filter-name
```

RELATED DOCUMENTATION

Overview of Firewall Filters (QFX Series)

Configuring Port Mirroring on SRX Series Firewalls

To configure port mirroring on an SRX device, you must first configure the `forwarding-options` and `interfaces` at the `[edit]` hierarchy level.

You must configure the `forwarding-options` statement to define an instance of the `mirror-to` port for port mirroring and also configure the interface to be mirrored.

> **NOTE**: The mirrored port and the mirror-to port must be under the same Broadcom chipset in an I/O card.

To configure port mirroring:

1. Specify the `rate` and `run-length` at the `[edit forwarding-options port-mirroring input]` hierarchy level:

   NOTE:

   - `rate`: Ratio of packets to be sampled (1 out of *N*) (1 through 65535)
   - `run-length`: Number of samples after initial trigger (0 through 20)

```
[edit]
    forwarding-options
        port-mirroring {
                input {
                    rate number;
                    run-length number;
                }
            }
```

2. To send the copies of the packet to the `mirror-to` port, include the `interface intf-name` statement at the `[edit forwarding-options port-mirroring family any output]` hierarchy level.

```
output {
    interface intf-name;
}
```

   NOTE: Port mirroring on SRX Series Firewalls uses `family any` to transfer the `mirror-to` port information to the Packet Forwarding Engine (PFE). The mirroring engine copies all the packets from `mirrored` port to the `mirror-to` port.

NOTE: You can configure an `instance` clause to specify multiple `mirror-to` ports.

To mirror an interface, include the `port-mirror-instance` statement at the `[edit interface mirrored-intf-name]` hierarchy level.

The mirrored interface is configured with an instance name, defined in the `forwarding-options`. The `mirrored` port and the `mirror-to` port are linked through that instance.

```
instance {
    inst-name {
        input {
            rate number;
            run-length number;
        }
        family any {
            output {
                interface intf-name;
            }
        }
    }
}
interfaces
        mirrored-intf-name {
            port-mirror-instance instance-name;
        }
```

NOTE: Port mirroring on SRX Series Firewalls does not differentiate the traffic direction, but mirrors the ingress and egress samples together.

A sample configuration for port mirroring is shown below:

```
mirror port ge-1/0/2 to port ge-1/0/9.0
forwarding-options
    port-mirroring {
        input {
            rate 1;
            run-length 10;
        }
        family any {
            output {
                interface ge-1/0/9.0;
            }
        }
        instance {
            inst1 {
                input {
```

```
                    rate 1;
                    run-length 10;
                }
                family any {
                    output {
                        interface ge-1/0/9.0;
                    }
                }
            }
        }
interfaces {
    ge-1/0/2  {
        port-mirror-instance inst1;
    }
}
```

Examples: Configuring Port Mirroring for Local Analysis

IN THIS SECTION

- Requirements | **1218**
- Overview and Topology | **1218**
- Example: Mirroring All Employee Traffic for Local Analysis | **1219**

Use port mirroring to send traffic to applications that analyze traffic for purposes such as monitoring compliance, enforcing policies, detecting intrusions, monitoring and predicting traffic patterns, correlating events, and so on. Port mirroring copies packets entering or exiting an interface or entering a VLAN and sends the copies to a local interface for local monitoring.

NOTE: This example uses the Enhanced Layer 2 Software (ELS) configuration style. For ELS details, see Using the Enhanced Layer 2 Software CLI.

This example describes how to configure port mirroring to copy traffic sent by employee computers to a switch to an access interface on the same switch.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 13.2
- A switch

Overview and Topology

IN THIS SECTION

- Topology | 1218

This topic includes two related examples that describe how to mirror traffic entering interfaces on the switch to an access interface on the same switch. The first example shows how to mirror all traffic sent by employee computers to the switch. The second example includes a filter to mirror only the employee traffic going to the Web.

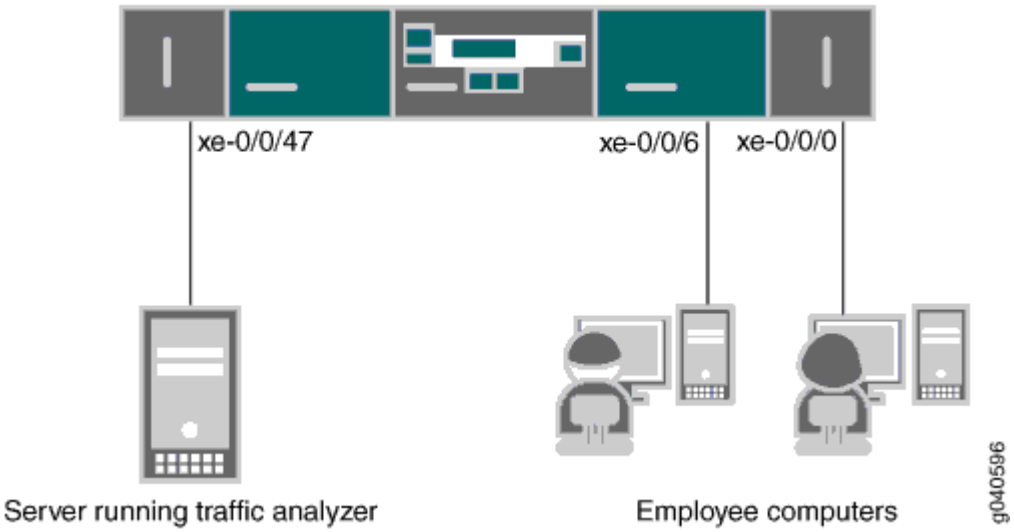
Topology

In this example, `xe-0/0/0` and `xe-0/0/6` serve as connections for employee computers. Interface `xe-0/0/47` is connected to a device running an analyzer application.

NOTE: Multiple ports mirrored to one interface can cause buffer overflow and dropped packets.

Figure 45 on page 1219 shows the network topology for this example.

Figure 45: Network Topology for Local Port Mirroring Example



Example: Mirroring All Employee Traffic for Local Analysis

IN THIS SECTION

- Procedure | 1219

To configure port mirroring for all traffic sent by employee computers for local analysis, perform the tasks explained in this section.

Procedure

CLI Quick Configuration

To quickly configure local port mirroring for ingress traffic to the two ports connected to employee computers, copy the following commands and paste them into a switch terminal window:

```
[edit]
 set interfaces xe-0/0/0 unit 0 family ethernet-switching
 set interfaces xe-0/0/6 unit 0 family ethernet-switching

 set interfaces xe-0/0/47 unit 0 family ethernet-switching
```

```
set forwarding-options analyzer employee-monitor input ingress interface xe-0/0/0.0
set forwarding-options analyzer employee-monitor input ingress interface xe-0/0/6.0
set forwarding-options analyzer employee-monitor output interface xe-0/0/47.0
```

Step-by-Step Procedure

To configure an analyzer called `employee-monitor` and specify the input (source) interfaces and the output interface:

1. Configure the interfaces connected to employee computers as input interfaces for the port-mirror analyzer `employee-monitor`:

```
[edit forwarding-options]
 user@switch# set analyzer employee-monitor input ingress interface xe-0/0/0.0
user@switch# set analyzer employee-monitor input ingress interface xe-0/0/6.0
```

2. Configure the output analyzer interface for the `employee-monitor` analyzer. This will be the destination interface for the mirrored packets:

```
[edit forwarding-options]
user@switch# set analyzer employee-monitor output interface xe-0/0/47.0
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show forwarding-options analyzer
    employee-monitor {
        input {
            ingress {
                interface xe-0/0/0.0;
                interface xe-0/0/6.0;
            }
        }
        output {
            interface {
                xe-0/0/47.0;
            }
```

```
        }
    }
}
```

Example: Mirroring Employee Web Traffic with a Firewall Filter

IN THIS SECTION

- Requirements | **1221**
- Overview | **1221**
- Configuring | **1221**
- Verification | **1225**

Requirements

This example uses the following hardware and software components:

- One QFX5100 switch
- Junos OS Release 14.1X53-D30

Overview

Rather than mirror all traffic, it is usually desirable to mirror only certain traffic. This is a more efficient use of your bandwidth and hardware and might be necessary due to constraints on these assets. To select specific traffic for mirroring, you use a firewall filter to match the desired traffic and direct it to a port-mirroring instance. The port-mirroring instance then copies the packets and sends them to the output VLAN, interface, or IP address.

Configuring

IN THIS SECTION

- Procedure | **1222**

To specify that the only traffic that will be mirrored is traffic sent by employees to the Web, perform the tasks explained in this section. To select this traffic for mirroring, you use a firewall filter to specify this traffic and direct it to a port-mirroring instance.

Procedure

CLI Quick Configuration

To quickly configure local port mirroring of traffic from employee computers that is destined for the Web, copy the following commands and paste them into a switch terminal window:

```
[edit]
 set interface xe-0/0/47 unit 0 family ethernet-switching
 set forwarding-options port-mirroring instance employee-web-monitor family ethernet-switching
output interface xe-0/0/47.0
 set firewall family ethernet-switching filter watch-employee term employee-to-corp from ip-
destination-address 192.0.2.16/28
 set firewall family ethernet-switching filter watch-employee term employee-to-corp from ip-
source-address 192.0.2.16/28
 set firewall family ethernet-switching filter watch-employee term employee-to-corp then accept
 set firewall family ethernet-switching filter watch-employee term employee-to-web from
destination-port 80
 set firewall family ethernet-switching filter watch-employee term employee-to-web then port-
mirror-instance employee-web-monitor
 set interfaces xe-0/0/0 unit 0 family ethernet-switching filter input watch-employee
 set interfaces xe-0/0/6 unit 0 family ethernet-switching filter input watch-employee
```

Step-by-Step Procedure

To configure local port mirroring of employee-to-web traffic from the two ports connected to employee computers:

1. Configure the output interface:

```
[edit interfaces]
user@switch# set xe-0/0/47 unit 0 family ethernet-switching
```

2. Configure the `employee-web-monitor` output interface. (Configure only the output—the input comes from the filter.)

```
[edit forwarding-options]
user@switch# set port-mirroring instance employee-web-monitor family ethernet-switching
output interface xe-0/0/47.0
```

3. Configure a firewall filter called `watch-employee` that includes a term to match traffic sent to the Web and send it to the port-mirroring instance `employee-web-monitor`. Traffic to and from the corporate subnet (destination or source address of `192.0.2.16/28`) does not need to be copied, so create another term to accept that traffic before it reaches the term that sends Web traffic to the instance:

```
[edit firewall family ethernet-switching]
user@switch# set filter watch-employee term employee-to-corp from ip-destination-address
192.0.2.16/28
user@switch# set filter watch-employee term employee-to-corp from ip-source-address
192.0.2.16/28
user@switch# set filter watch-employee term employee-to-corp then accept
user@switch# set filter watch-employee term employee-to-web from destination-port 80
user@switch# set filter watch-employee term employee-to-web then port-mirror-instance
employee-web-monitor
```

4. Apply the firewall filter to the appropriate interfaces as an ingress filter (egress filters do not allow analyzers):

```
[edit interfaces]
user@switch# set xe-0/0/0 unit 0 family ethernet-switching filter input watch-employee
user@switch# set xe-0/0/6 unit 0 family ethernet-switching filter input watch-employee
```

Results

Check the results of the configuration:

```
[edit]
user@switch# show
forwarding-options {
    port-mirroring {
        instance     {
```

```
            employee-web-monitor    {
                family ethernet-switching {
                        output {
                            interface xe-0/0/47.0;
                        }
                }
            }
        }
    }
}
...
firewall {
    family ethernet-switching {
        filter watch-employee {
            term employee-to-corp {
            from {
                    ip-source-address 192.0.2.16/28;
                    ip-destination-address 192.0.2.16/28;
                }
                then accept;
            term employee-to-web {
                from {
                    destination-port 80;
                }
                then port-mirror-instance employee-web-monitor;
            }
        }
    }
}
...
interfaces {
    xe-0/0/0 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
    xe-0/0/6 {
        family ethernet-switching {
            filter {
```

```
                input watch-employee;
            }
        }
    }
    xe-0/0/47 {
        family ethernet-switching;
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1225**

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the port-mirroring instance named `employee-web-monitor` has been created on the switch with the appropriate input interfaces and appropriate output interface.

Action

You can verify that the port mirror port-mirroring instance has been configured as expected by using the `show forwarding-options port-mirroring` command.

```
user@switch> show forwarding-options port-mirroring
Instance name                 : employee-web-monitor
Instance Id:  2
Input parameters:
    Rate                              :1
    Run-length                        :0
    Maximum packet length             :0
Output parameters:
    Family                State       Destination      Next-hop
```

```
    ethernet-switching     up         xe-0/0/47.0
```

Meaning

This output shows the following information about the port-mirroring instance `employee-web-monitor`:

- Has a rate of `1` (mirroring every packet, the default setting)
- The number of consecutive packets sampled (run-length) is `0`
- The maximum size of the original packet that was mirrored is `0` (`0` indicates the entire packet)
- The state of the output parameters: `up` indicates that the instance is mirroring the traffic entering the xe-0/0/0 and xe-0/0/6 interfaces, and is sending the mirrored traffic to the xe-0/0/47 interface

If the state of the output interface is `down` or if the output interface is not configured, the `state` value will be `down` and the instance will not be programmed for mirroring.

Example: Configuring Port Mirroring for Remote Analysis

IN THIS SECTION

- Requirements | **1226**
- Overview and Topology | **1227**
- Mirroring All Employee Traffic for Remote Analysis | **1227**
- Mirroring Employee-to-Web Traffic for Remote Analysis | **1229**
- Verification | **1233**

Use port mirroring to send traffic to applications that analyze traffic for purposes such as monitoring compliance, enforcing policies, detecting intrusions, monitoring and predicting traffic patterns, correlating events, and so on. Port mirroring copies packets entering or exiting an interface or entering a VLAN and sends the copies either to a local interface for local monitoring or to a VLAN for remote monitoring. This example describes how to configure port mirroring for remote analysis.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 13.2 for the QFX Series

- A switch

Overview and Topology

IN THIS SECTION

- Topology | 1227

This topic includes two related examples that describe how to mirror traffic entering ports on the switch to an analyzer VLAN so that you can perform analysis using a remote device. The first example shows how to mirror all traffic sent by employee computers to the switch. The second example includes a filter to mirror only the employee traffic going to the Web.

Topology

In this example:

- Interfaces `ge-0/0/0` and `ge-0/0/1` are Layer 2 interfaces that connect to employee computers.
- Interface `ge-0/0/2` is a Layer 2 interface that connects to another switch.
- VLAN `remote-analyzer` is configured on all switches in the topology to carry the mirrored traffic.

NOTE: In addition to performing the configuration steps described here, you must also configure the analyzer VLAN (`remote-analyzer` in this example) on the other switches that are used to connect the source switch (the one in this configuration) to the one that the monitoring station is connected to.

Mirroring All Employee Traffic for Remote Analysis

IN THIS SECTION

- Procedure | 1228

Procedure

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `edit` hierarchy level:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching port-mode trunk
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
 set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/0.0
 set forwarding-options analyzer employee-monitor input ingress interface ge-0/0/1.0
 set forwarding-options analyzer employee-monitor output vlan remote-analyzer
```

Step-by-Step Procedure

To configure basic remote port mirroring:

1. Configure the analyzer VLAN (called `remote-analyzer` in this example):

   ```
   [edit vlans]
   user@switch# set vlans remote-analyzer vlan-id 999
   ```

2. Configure the interface connected to another switch for trunk mode and associate it with the `remote-analyzer` VLAN:

   ```
   [edit interfaces]
   user@switch# set ge-0/0/10 unit 0 family ethernet-switching port-mode trunk
   user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
   ```

3. Configure the `employee-monitor` analyzer:

   ```
   [edit forwarding-options]
   user@switch# set analyzer employee-monitor
   user@switch# set analyzer employee-monitor input ingress interface ge-0/0/0.0
   ```

```
user@switch# set analyzer employee-monitor input ingress interface ge-0/0/1.0
user@switch# set analyzer employee-monitor output vlan remote-analyzer
```

4. Configure the `remote-analyzer` VLAN on the switches that connect this switch to the monitoring workstation.

Results

Check the results of the configuration:

```
[edit]
user@switch# show
 forwarding-options {
    analyzer employee-monitor {
        input {
            ingress {
                interface ge-0/0/0.0;
                interface ge-0/0/1.0;
            }
        }
        output {
            vlan {
                remote-analyzer;
            }
        }
    }
}
```

Mirroring Employee-to-Web Traffic for Remote Analysis

IN THIS SECTION

- CLI Quick Configuration | 1230
- Procedure | 1230

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `edit` hierarchy level:

```
[edit]
 set vlans remote-analyzer vlan-id 999
 set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
 set interfaces ge-0/0/10 unit 0 family ethernet-switching vlan members 999
 set forwarding-options port-mirroring instance employee-web-monitor loss-priority high output
vlan 999
 set firewall family ethernet-switching filter watch-employee term employee-to-web from
destination-port 80
 set firewall family ethernet-switching filter watch-employee term employee-to-web then port-
mirror-instance employee-web-monitor
 set ge-0/0/0 unit 0 family ethernet-switching filter input watch-employee
 set interfaces ge-0/0/1 unit 0 family ethernet-switching filter input watch-employee
```

Procedure

Step-by-Step Procedure

1. Configure the analyzer VLAN (called `remote-analyzer` in this example):

```
[edit vlans]
user@switch# set remote-analyzer vlan-id 999
```

2. Configure an interface to associate it with the `remote-analyzer` VLAN:

```
[edit interfaces]
user@switch# set interfaces ge-0/0/10 unit 0 family ethernet-switching interface-mode trunk
user@switch# set ge-0/0/10 unit 0 family ethernet-switching vlan members 999
```

3. Configure the `employee-web-monitor` analyzer. (Configure only the output—the input comes from the filter.)

```
[edit forwarding-options]
user@switch# set forwarding-options port-mirroring instance employee-web-monitor output vlan
999
```

4. Configure a firewall filter called `watch-employee` to match traffic sent to the Web and send it to the analyzer `employee-web-monitor`:

```
[edit firewall family ethernet-switching]
user@switch# set filter watch-employee term employee-to-web from destination-port 80
user@switch# set filter watch-employee term employee-to-web then port-mirror-instance
employee-web-monitor
```

5. Apply the firewall filter to the appropriate interfaces as an ingress filter:

```
[edit interfaces]
user@switch# set ge-0/0/0 unit 0 family ethernet-switching filterinput watch-employee
user@switch# set ge-0/0/1 unit 0 family ethernet-switching filter input watch-employee
```

6. Configure the `remote-analyzer` VLAN on the switches that connect this switch to the monitoring workstation.

Results

Check the results of the configuration:

```
[edit]
user@switch# show
interfaces {
    ...
    ge-0/0/10 {
        unit 0 {
            family ethernet-switching {
                interface-mode trunk;
                vlan {
                    members remote-analyzer;
                }
```

```
            }
        }
    }
    ge-0/0/0 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
    ge-0/0/1 {
        unit 0 {
            family ethernet-switching {
                filter {
                    input watch-employee;
                }
            }
        }
    }
}
...
firewall {
     family ethernet-switching {
        ...
        filter watch-employee {
            term employee-to-web {
                from {
                    destination-port 80;
                }
                then port-mirror-instance employee-web-monitor;
            }
        }
    }
}
 forwarding-options analyzer {
    employee-web-monitor {
        output {
            vlan {
                999;
            }
        }
```

```
    }
vlans {
    remote-analyzer {
        vlan-id 999;
    }
}
```

Verification

IN THIS SECTION

- Verifying That the Analyzer Has Been Correctly Created | **1233**

Verifying That the Analyzer Has Been Correctly Created

Purpose

Verify that the analyzer named `employee-monitor` or `employee-web-monitor` has been created on the switch with the appropriate input interfaces and appropriate output interface.

Action

You can verify the port mirror analyzer is configured as expected using the `show analyzer` command.

```
user@switch> show analyzer
    Analyzer name                : employee-monitor
    Output VLAN                  : remote-analyzer
    Ingress monitored interfaces : ge-0/0/0.0
    Ingress monitored interfaces : ge-0/0/1.0
```

Meaning

This output shows that the `employee-monitor` analyzer is mirroring the traffic entering `ge-0/0/0` and `ge-0/0/1` and is sending the mirror traffic to the analyzer `remote-analyzer`.

1:N Port Mirroring to Multiple Destinations on Switches

SUMMARY

You can use the port mirroring feature described in this document to mirror traffic to multiple Layer 2 destinations.

IN THIS SECTION

- 1:N Port Mirroring—Description and Configuration Guidelines | **1234**
- Configure the Port-Mirroring Instance | **1236**
- Configure the Native Analyzer | **1237**
- Configure Next-Hop Groups | **1237**
- Configure the Firewall Filter | **1237**
- Configure the Interfaces | **1237**
- Configure the VLANs | **1237**
- Sample Configuration Results | **1238**

1:N Port Mirroring—Description and Configuration Guidelines

IN THIS SECTION

- What Is 1:N Port Mirroring? | **1234**
- Getting Ready to Configure 1:N Port Mirroring—Guidelines and Limitations | **1235**
- Overview of Configuration Tasks for 1:N Port Mirroring | **1236**

What Is 1:N Port Mirroring?

We use the term *1:N port mirroring* in this document to refer to the feature that enables you to mirror packets to multiple destinations. "1" represents the packet source being mirrored and "N" represents the multiple destinations the packet is sent to. You might also see this feature described as *multipacket mirroring*.

Port mirroring helps network administrators to debug network problems and to fend off attacks on the network. You can use port mirroring for traffic analysis on network devices such as routers and switches that, unlike hubs, do not broadcast packets to every interface on the destination device. Port mirroring sends copies of all packets to local or remote analyzers where you can monitor and analyze the data.

You use 1:N port mirroring to mirror traffic to multiple Layer 2 destinations. You use next-hop groups in this feature configuration.

You configure these multiple observing ports with connections to different monitoring devices.

Getting Ready to Configure 1:N Port Mirroring—Guidelines and Limitations

You can configure the 1:N port mirroring feature in the following two configuration methods:

- Port mirroring (using a firewall filter-based method) at the `[edit forwarding-options port-mirroring instance]` hierarchy
- Native analyzer at the `[edit forwarding-options analyzer]` hierarchy

> **NOTE**: You can configure both of the preceding methods on the same device. See "Sample Configuration Results" on page 1238 for an example.

The following address families are supported in 1:N port mirroring:

- `ethernet-switching`
- `inet`
- `inet6`

Here are the **limitations** that you need to keep in mind as you configure the feature:

- Next-hop group members can be Layer 2 only, not Layer 3.
- You can configure `next-hop-group output` support only for **local** port mirroring—that is, not for remote port mirroring or for remote port mirroring to an IP address (GRE encapsulation).
- You can configure as many as 4 next-hop groups, and you can add up to 4 interfaces to each next-hop group. You must define at least 2 destinations to send packets to more than one destination; however, you can define just one destination in a next-hop group.

Table 124 on page 1235 lists the **configuration-hierarchy combinations** you use to build your 1:N mirroring topology:

Table 124: Configuration Hierarchies for 1:N Port Mirroring

Configuration Method	Hierarchies
Port mirroring (filter-based)	`[edit forwarding-options port-mirroring instance]`

Table 124: Configuration Hierarchies for 1:N Port Mirroring *(Continued)*

Configuration Method	Hierarchies
	[edit firewall family *family-name* filter]
	[edit forwarding-options next-hop-group]
	[edit interfaces]
	[edit vlans]
Native analyzer	[edit forwarding-options analyzer]
	[edit forwarding-options next-hop-group]
	[edit interfaces]
	[edit vlans]

NOTE: You can read through the configuration task subsections, or you can jump to the "Sample Configuration Results" on page 1238 that shows the combined task results.

Overview of Configuration Tasks for 1:N Port Mirroring

The following configuration task subsections show you how to configure each of the hierarchies listed in Table 1 on page 1235. You can read through the configuration task subsections, or you can jump to the "Sample Configuration Results" on page 1238 that shows the combined task results.

Configure the Port-Mirroring Instance

To configure the port-mirroring instance, enter the following commands in the configuration mode [edit]:

set forwarding-options port-mirroring instance *instance-name* family *family-name* output next-hop-group *next-hop-group-name*

Configure the Native Analyzer

To configure the native analyzer, enter the following commands in the configuration mode `[edit]`:

1. set forwarding-options analyzer *analyzer-name* input ingress interface *interface-name*
2. set forwarding-options analyzer *analyzer-name* output next-hop-group *next-hop-group-name*

Configure Next-Hop Groups

To configure next-hop groups, enter the following command in the configuration mode `[edit]`:

> **NOTE**: You must configure the `group-type` value as `layer-2`.

set forwarding-options next-hop-group *next-hop-group-name* group-type layer-2 interface *interface-name*

Configure the Firewall Filter

To configure the firewall filter, enter the following commands in the configuration mode `[edit]`:

> **NOTE**: Define a firewall filter that references the next-hop group as the filter action.

For information about configuring firewall filters in general, see the Routing Policies, Firewall Filters, and Traffic Policers User Guide.

1. set firewall family *family-name* filter *filter-name* term *term-name* then port-mirror-instance *instance-name*
2. set firewall family *family-name* filter *filter-name* term *term-name* from source-port *port-number*

Configure the Interfaces

To configure the interfaces, enter the following commands in the configuration mode `[edit]`:

1. set interfaces *interface-name* unit *logical-unit-number* family *family-name* interface-mode *mode*
2. set interfaces *interface-name* unit *logical-unit-number* family *family-name* filter input *filter-name*

Configure the VLANs

To configure VLANs, enter the following commands in the configuration mode `[edit]`:

set vlans *vlan-name* vlan-id *vlan-id*

Sample Configuration Results

```
set interfaces ge-2/1/9 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-2/1/9 unit 0 family ethernet-switching vlan members 100-102
set interfaces ge-2/2/7 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-2/2/7 unit 0 family ethernet-switching vlan members 100-102
set interfaces ge-2/3/0 unit 0 family ethernet-switching interface-mode trunk
set interfaces ge-2/3/0 unit 0 family ethernet-switching vlan members 100-102
set interfaces ge-2/3/0 unit 0 family ethernet-switching filter input f1
set forwarding-options analyzer analyz1 input ingress interface ge-2/3/0.0
set forwarding-options analyzer analyz1 output next-hop-group nhg1
set forwarding-options port-mirroring instance inst1 family ethernet-switching output next-hop-
group
nhg1
set forwarding-options next-hop-group nhg1 group-type layer-2
set forwarding-options next-hop-group nhg1 interface ge-2/2/7.0
set firewall family ethernet-switching filter f1 term t1 from source-port 7023
set firewall family ethernet-switching filter f1 term t1 then port-mirror-instance inst1
```

Monitoring Port Mirroring

IN THIS SECTION

- Displaying Layer 2 Port-Mirroring Instance Settings and Status | **1238**
- Displaying Next-Hop Group Settings and Status | **1239**

Displaying Layer 2 Port-Mirroring Instance Settings and Status

To display the current state of port-mirroring instances, use the `show forwarding-options port-mirroring <terse | detail> <instance-name>` operational command.

For more information about displaying port mirroring instance settings and status, see the Junos OS Administration Library.

Displaying Next-Hop Group Settings and Status

To display the current state of next-hop groups, use the `show forwarding-options next-hop-group <terse | brief | detail> <group-name>` operational command.

For more information, see the CLI Explorer.

Configure Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic

SUMMARY

Selective packet mirroring filters can serve as a highly effective troubleshooting mechanism and can also be used for performance monitoring purposes.

IN THIS SECTION

- Understanding Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic | **1239**
- Configure a Filter with a Port-Mirroring Instance or with Global Port Mirroring | **1240**
- Configure Mirroring for FTI Tunnels | **1243**
- Attachment Points for Filters | **1247**
- Suggestions for Enhancements to Your Packet-Filtering Configuration | **1247**

Understanding Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic

IN THIS SECTION

- Features of Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic | **1240**
- Limitations for the Packet-Level Mirroring Configuration | **1240**

This document focuses on a capability to select traffic using a wide variety of IPv4 or IPv6 filter match conditions and to mirror entire packets with their original Layer 2 header information.

Layer 2 header information might be essential to identify a specific customer in an edge router deployment or a specific Internet peer in a public peering case.

Features of Packet Mirroring with Layer 2 Headers for Layer 3 Forwarded Traffic

In a nutshell, you can mirror the original Layer 2 packet header when the `l2-mirror` action is configured in a `family inet` or `family inet6` filter. Packets can be mirrored locally or remotely by using GRE tunnels.

If you specify the output interface in your mirroring configuration as a GRE tunnel interface, packets are encapsulated in GRE before transmission. A port-mirroring instance can be configured with multiple output protocol families.

Limitations for the Packet-Level Mirroring Configuration

- The new action, `l2-mirror`, is only supported for `family inet` and `family inet6`filters.
- Layer 2 mirroring is not supported on gr-*/*/* interfaces.

Configure a Filter with a Port-Mirroring Instance or with Global Port Mirroring

You configure `l2-mirror` under either `firewall family (inet | inet6) filter` *`filter-name`* `term then port-mirror` (global port mirroring) or `firewall (inet | inet6) filter` *`filter-name`* `term then port-mirror-instance` *`instance-name`* (port-mirroring instances, or "PM instances").

Having `l2-mirror` configured for a term indicates that for packets matching this term, the Layer 2 packet is mirrored. The software performs commit checks for invalid configurations, such as when `l2-mirror` is configured but no port-mirroring output interface is configured for `family any` in the global-level or instance-level port mirroring configuration. If you deactivate `l2-mirror`, the mirroring behavior reverts to Layer 3 mirroring.

The following two examples show the configuration of a filter (the filter name in the examples is f1) with a port-mirroring instance and with global port mirroring. In both examples traffic is mirrored to the remote destination over a GRE tunnel.

> **NOTE**: The port-mirroring configurations, which are under `forwarding-options`, are configured with `family any`, but the match conditions in the filter configuration are done under `family inet`. Using `family any` enables the mirroring of Layer 2 packets.

1. To configure the filter with a **port-mirroring instance**:

   > **NOTE**: You can specify a gr- interface as your mirror destination. See Configuring Generic Routing Encapsulation Tunneling on ACX Series for information on configuring gr- interfaces

(the document refers specifically to ACX Series routers; the same information applies to various other routers, including MX10003.)

```
forwarding-options {
  port-mirroring {
    instance {
      mirror-instance-1 {
        input {
          rate 2;
        }
        family any {
          output {
            interface gr-0/0/0.0;
          }
        }
      }
    }
  }
}
firewall {
  family inet {
    filter f1 {
      term tcp-flags {
        from {
          protocol tcp;
          tcp-flags "(syn & fin & rst)";
        }
        then {
          port-mirror-instance mirror-instance-1;
          l2-mirror;
        }
      }
    }
  }
}
interfaces {
    gr-0/0/0 {
        unit 0 {
            tunnel {
                source 10.1.1.2/32;
                destination 10.1.1.1/32;
```

```
            }
            family bridge {
                interface-mode access;
                vlan-id 100;
            }
        }
    }
}
routing-instances {
    i1 {
        instance-type virtual-switch;
        interface gr-0/0/0.0;
        bridge-domains {
            bd100 {
                vlan-id 100;
            }
        }
    }
}
```

2. To configure the filter with **global port mirroring**:

```
forwarding-options {
  port-mirroring {
    input {
      rate 2;
    }
    family any {
      output {
        interface gr-0/0/0.0;
      }
    }
  }
}
firewall {
  family inet {
    filter f1 {
      term tcp-flags {
        from {
          protocol tcp;
          tcp-flags "(syn & fin & rst)";
        }
```

```
        then {
          port-mirror;
          l2-mirror;
        }
      }
    }
  }
}
interfaces {
    gr-0/0/0 {
        unit 0 {
            tunnel {
                source 10.1.1.2/32;
                destination 10.1.1.1/32;
            }
            family bridge {
                interface-mode access;
                vlan-id 100;
            }
        }
    }
}
routing-instances {
    i1 {
        instance-type virtual-switch;
        interface gr-0/0/0.0;
        bridge-domains {
            bd100 {
                vlan-id 100;
            }
        }
    }
}
```

Configure Mirroring for FTI Tunnels

When the data path traverses a flexible tunnel interface (FTI) tunnel, the output packet is sent with tunnel encapsulation. You can set up a configuration that mirrors the original packet as well as the packet with all encapsulations as it egresses out.

To mirror the original packet, configure input mirroring on the ingress WAN interface.

To mirror the packet with all encapsulations, enable output mirroring on the egress WAN interface.

To enable mirroring based on a filter installed on the FTI interface, you use a two-step process:

1. You mark packets for mirroring using the policy action at the fti- interface. The policy action is typically used to select the egress rewrite rule, but in this case, the policy action is used to mark interesting packets with an internal policy attribute, without any special rewrite rule configured.
2. You have the software intercept packets that match the specific policy on the egress WAN side and initiate the `l2-mirror` action. Packets are reported with Layer 2 header information, including tunnel encapsulation.

NOTE: The following example shows Layer 3 port mirroring. To obtain Layer 2 port mirroring, simply configure the `l2-mirror` action as shown in the preceding examples in this document.

1. Define `policy-map` *`policy-map-name`* under the `class-of-service` stanza:

```
class-of-service {
  policy-map {
    pm1;
  }
}
```

2. Apply an output filter on the FTI with action `policy-map pm1`:

```
family inet {
  filter mirror-all {
    term mirror {
      from {
        policy-map pm1;
      }
      then {
        count all;
        port-mirror-instance mirror-to-gre;
        accept;
      }
    }
     term default {
       then accept;
     }
   }
  filter f1 {
    term t1 {
```

```
      from {
        source-address {
          10.1.1.2/32;
        }
      }
      then {
        policy-map pm1;
        count c1;
      }
    }
    term t2 {
      from {
        source-address {
          10.36.100.1/32;
        }
      }
      then accept;
    }
  }
}
```

3. The following configuration output shows the FTI configuration on interface fti0.1001. (For more detail on configuring an FTI tunnel, see Flexible Tunnel Interfaces Overview.)

```
interfaces {
  fti0 {
    unit 1001 {
     tunnel {
       encapsulation vxlan-gpe {
         source {
           address 198.51.100.1;
         }
         destination {
           address 198.51.100.2;
         }
         tunnel-endpoint vxlan;
         destination-udp-port 4789;
         vni 22701;
       }
     }
     family inet {
       filter {
```

```
          output f1;
        }
        address 10.18.1.1/27;
      }
      family inet6 {
        address 2001:db8::1:1/126;
      }
    }
  }
}
```

4. Add a filter (here named `mirror-all`) on the egress WAN interface with match from `policy-map pm1 then port-mirror`:

```
family inet {
  filter mirror-all {
    term mirror {
      from {
        policy-map policy-map-name;
      }
      then {
        count all;
        port-mirror-instance mirror-to-gre;
                accept;
      }
    }
    term default {
      then accept;
    }
  }
}
```

```
interfaces {
 ge-0/0/1 {
  unit 0 {
    family inet {
      filter {
        output mirror-all;
      }
      address 10.200.0.1/24;
    }
```

```
    family iso;
  }
 }
}
```

Attachment Points for Filters

Filter Attachment Point	Interface Type	Mirrored Packet Layer 2 Header
Input	Any Ethernet except gr- and fti-	Layer 2 header of the incoming packet is reported
Output	Any Ethernet except gr- and fti-	Layer 2 header of the incoming packet is reported
Input or output	gr- interface	Not supported
Input	fti- interface	Incoming Layer 2 header of the original packet (as it was seen on the WAN port)
Output	fti- interface	Incoming Layer 2 header of the original packet (as it was seen on the WAN port)
Input	irb interface	Incoming Layer 2 header of the original packet (as it was seen on the WAN port)
Output	irb interface	Not supported

Suggestions for Enhancements to Your Packet-Filtering Configuration

Consider the following as an additional practice to enhance your filter network telemetry setup:

You can use input-chain and output-chain filters to separate the filter configuration used for mirroring from existing filters, thus helping you to avoid inadvertent configuration errors while troubleshooting. For details of this feature, see Example: Using Firewall Filter Chains.

Troubleshooting Port Mirroring

IN THIS SECTION

- Troubleshooting Port Mirroring | 1248
- Troubleshooting Port Mirroring Configuration Error Messages | 1250

Troubleshooting Port Mirroring

IN THIS SECTION

- Egress Port Mirroring with VLAN Translation | 1248
- Egress Port Mirroring with Private VLANs | 1249

Egress Port Mirroring with VLAN Translation

IN THIS SECTION

- Problem | 1248
- Solution | 1249

Problem

Description

If you create a port-mirroring configuration that mirrors customer VLAN (CVLAN) traffic on egress and the traffic undergoes VLAN translation before being mirrored, the VLAN translation does not apply to the mirrored packets. That is, the mirrored packets retain the service VLAN (SVLAN) tag that should be replaced by the CVLAN tag on egress. The original packets are unaffected—on these packets VLAN translation works properly, and the SVLAN tag is replaced with the CVLAN tag on egress.

Solution

This is expected behavior.

SEE ALSO

Understanding Q-in-Q Tunneling and VLAN Translation

Egress Port Mirroring with Private VLANs

IN THIS SECTION

Problem | 1249
Solution | 1250

Problem

Description

If you create a port-mirroring configuration that mirrors private VLAN (PVLAN) traffic on egress, the mirrored traffic (the traffic that is sent to the analyzer system) has the VLAN tag of the ingress VLAN instead of the egress VLAN. For example, assume the following PVLAN configuration:

- Promiscuous trunk port that carries primary VLANs pvlan100 and pvlan400.
- Isolated access port that carries secondary VLAN isolated200. This VLAN is a member of primary VLAN pvlan100.
- Community port that carries secondary VLAN comm300. This VLAN is also a member of primary VLAN pvlan100.
- Output interface (monitor interface) that connects to the analyzer system. This interface forwards the mirrored traffic to the analyzer.

If a packet for pvlan100 enters on the promiscuous trunk port and exits on the isolated access port, the original packet is untagged on egress because it is exiting on an access port. However, the mirror copy retains the tag for pvlan100 when it is sent to the analyzer.

Here is another example: If a packet for comm300 ingresses on the community port and egresses on the promiscuous trunk port, the original packet carries the tag for pvlan100 on egress, as expected. However, the mirrored copy retains the tag for comm300 when it is sent to the analyzer.

Solution

This is expected behavior.

Troubleshooting Port Mirroring Configuration Error Messages

IN THIS SECTION

- An Analyzer Configuration Returns a “Multiple interfaces cannot be configured as a member of Analyzer output VLAN” Error Message | **1250**

Troubleshooting issues with port mirroring on EX Series switches:

An Analyzer Configuration Returns a “Multiple interfaces cannot be configured as a member of Analyzer output VLAN” Error Message

IN THIS SECTION

- Problem | **1250**
- Solution | **1251**

Problem

Description

In an analyzer configuration, if the VLAN to which mirrored traffic is sent contains more than one member interface, the following error message is displayed in the CLI when you commit the analyzer configuration and the commit fails:

```
Multiple interfaces cannot be configured as a member of Analyzer output VLAN <vlan name>
```

Solution

You must direct the mirrored traffic to a VLAN that has a single member interface. You can do this by completing either of these tasks:

- Reconfigure the existing VLAN to contain a single member interface. You can choose this method if you want to use the existing VLAN.
- Create a new VLAN with a single member interface and associate the VLAN with the analyzer.

To reconfigure the existing VLAN to contain only one member interface:

1. Remove member interfaces from the VLAN repeatedly by using either the `delete vlan` command or the `delete interface` command until the VLAN contains a single member interface:

   - ```
     [edit]
     user@switch# delete vlan vlan-id interface interface-name
     ```

   - ```
     [edit]
     user@switch# delete interface interface-name unit 0 family family-name vlan member vlan-id
     ```

2. (Optional) Confirm that the VLAN contains only one interface:

   ```
   [edit]
   user@switch# show vlans vlan-name
   ```

   The output for this command must display only one interface.

To create a new VLAN with a single member interface:

1. Configure a VLAN to carry the mirrored traffic:

   ```
   [edit]
   user@switch# set vlans vlan-name
   ```

2. Associate an interface with the VLAN:

   ```
   [edit]
   user@switch# set interfaces interface-name unit logical-unit-number family family-name vlan
   members vlan-name
   ```

3. Associate the VLAN with the analyzer:

```
[edit ethernet-switching-options]
user@switch# set analyzer analyzer-name output vlan vlan-name
```

PART 10

System Log Messages

Overview of System Logging | 1254

System Logging on a Single-Chassis System | 1266

System Logging for a TX Matrix or TX Matrix Plus Router | 1289

Direct System Log Messages to a Remote Destination | 1310

Display System Log Files | 1325

Configure System Logging for Security Devices | 1330

Configure Syslog over TLS | 1362

Monitor Log Messages | 1372

Overview of System Logging

SUMMARY

This section describes the system log messages that identify the Junos OS process responsible for generating the message and provides a brief description of the operation or error that occurred.

IN THIS SECTION

- System Log Overview | **1254**
- System Logging Facilities and Message Severity Levels | **1256**
- Default System Log Settings | **1258**
- Platform-Specific Default System Log Messages | **1260**
- Interpret Messages Generated in Standard Format | **1261**
- Manage Host OS System Log and Core Files | **1263**

System Log Overview

Junos OS generates system log messages (also called *syslog messages*) to record events that occur on the device, including the following:

- Routine operations, such as creation of an Open Shortest Path First (*OSPF*) protocol adjacency or a user login to the configuration database.
- Failure and error conditions, such as failure to access a configuration file or unexpected closure of a connection to a peer process.
- Emergency or critical conditions, such as power-down of the device due to excessive temperature.

Each system log message identifies the Junos OS process responsible for generating the message and provides a brief description of the operation or error that occurred. For detailed information about specific system log messages, see the System Log Explorer.

To configure the device to log system messages, configure the *syslog* statement at the [edit system] hierarchy level.

In Junos OS Release 17.3R1, the syslog-event daemon handles the fxp0 in dedicated management routing instance for IPv4 addressed remote host. In Junos OS Release 18.1R1, the syslog-event daemon supports IPv6-based configuration when connecting to a remote host or an archival site and fxp0 is

moved to dedicated management instance. In Junos OS Release 18.4R1, the syslog client can send messages through any routing instance that you define at appropriate hierarchies. See *routing-instance (Syslog)*.

NOTE: This topic describes system log messages for Junos OS processes and libraries and not the system logging services on a *Physical Interface Card* (*PIC*) such as the Adaptive Services PIC.

In Junos OS Evolved, each node has the standard `journalctl` tool, which is an interface to retrieve and filter the system journal. System log messages are extracted from the system journal. The `relay-eventd` process runs on all nodes and retrieves events (based on the syslog configuration) from the system journal as well as error messages from the different applications and forwards them to the `master-eventd` process. The `master-eventd` process runs on the primary Routing Engine and writes the log messages and errors to disk.

Use the System Log Explorer application to view or compare system log messages in different releases.

In Junos OS Evolved there is no `messages` file on the backup Routing Engine. All backup Routing Engine logs are in the `messages` file on the primary Routing Engine node.

By default, Junos OS Evolved appends the node name to the hostname in system log messages; Junos OS does not. This action keeps Junos OS Evolved system log messages compliant with RFC5424. However, some monitoring systems may not identify a Junos OS Evolved hostname correctly, because the hostname-node name combination does not match any hostnames in the inventory of hostnames.

Starting in Junos OS Evolved Release 20.4R2, to ensure accurate identification of Junos OS Evolved hostnames in your monitoring system, use the `set system syslog alternate-format` configuration mode command. This command changes the format of the Junos OS Evolved system log messages. The node name is prepended to the process name in the message rather than appended to the hostname, thereby allowing the monitoring system to identify the hostname correctly.

For example, Junos OS system log messages do not print the origin process in system log messages coming from an FPC:

```
user@mxhost> show log messages
Dec 19 13:22:41.959  mxhost chassisd[5290]: CHASSISD_IFDEV_DETACH_FPC: ifdev_detach_fpc(0)
Dec 19 13:23:22.900  mxhost  fpc2  Ukern event counter Sock_tx init delayed
```

However, Junos OS Evolved messages append the node name to the hostname and do print the origin process for messages coming from a node, including FPCs:

```
user@ptxhost-re0> show log messages
May 25 18:41:05.375  ptxhost-re0  mgd[16201]:  UI_CHILD_STATUS: Cleanup child '/usr/sbin/
```

```
dot1xd', PID 21322, status 0
May 25 18:42:34.632  ptxhost-fpc0  evo-cda-bt[14299]:  Register bt.igp_misc.debug.hdr_length_cnt
not found
May 25 18:42:34.753  ptxhost-fpc1  evo-cda-bt[14427]:  HBM: hbm_gf_register_inst
May 25 18:47:14.498  ptxhost-re0  ehmd[5598]:  SYSTEM_APP_READY: App is ready re0-ehmd
```

If you have configured the alternate format for Junos OS Evolved system log messages, the same set of system log messages would look like this instead, with the hostname by itself:

```
user@ptxhost-re0> show log messages
May 25 18:41:05.375  ptxhost re0- mgd[16201]:  UI_CHILD_STATUS: Cleanup child '/usr/sbin/
dot1xd', PID 21322, status 0
May 25 18:42:34.632  ptxhost fpc0- evo-cda-bt[14299]:  Register bt.igp_misc.debug.hdr_length_cnt
not found
May 25 18:42:34.753  ptxhost fpc1- evo-cda-bt[14427]:  HBM: hbm_gf_register_inst
May 25 18:47:14.498  ptxhost re0- ehmd[5598]:  SYSTEM_APP_READY: App is ready re0-ehmd
```

Starting in Junos OS Release 22.1R1 on SRX Series and NFX Series devices and Junos OS Evolved Release 22.2R1 on QFX5130, QFX5200, QFX5220, and QFX5700 devices, we've added multiple events inside the event tag using the <event>UI_LOGIN_EVENT|UI_LOGOUT_EVENT</event> format, which has an option (|) to separate the events and to generate system log messages. Earlier to these releases, the event tag used the <event>UI_LOGIN_EVENT UI_LOGOUT_EVENT</event> format and for various combinations of <get-syslog-events> rpc filters was not getting logged.

System Logging Facilities and Message Severity Levels

Table 125 on page 1256 lists the Junos OS system logging facilities that you can specify in configuration statements at the [edit system syslog] hierarchy level.

Table 125: Junos OS System Logging Facilities

Facility (number)	Type of Event or Error
kernel (0)	The Junos OS kernel performs actions and encounters errors.
user (1)	User-space perform actions or encounter errors.

Table 125: Junos OS System Logging Facilities *(Continued)*

Facility (number)	Type of Event or Error
daemon (3)	System perform actions or encounter errors.
authorization (4)	Authentication and authorization attempts.
ftp (11)	FTP performs actions or encounters errors.
ntp (12)	Network Time Protocol performs actions or encounters errors.
security (13)	Security related events or errors.
dfc (17)	Events related to dynamic flow capture.
external (18)	The local external applications perform actions or encounter errors.
firewall (19)	The firewall filter performs packet filtering actions.
pfe (20)	The Packet Forwarding Engine performs actions or encounters errors.
conflict-log (21)	Specified configuration is invalid on the router type.
change-log (22)	Changes to the Junos OS configuration.
interactive-commands (23)	A client application such as a Junos XML protocol or NETCONF XML client issues commands at the Junos OS command-line interface (CLI) prompt.

Table 126 on page 1258 lists the severity levels that you can specify in configuration statements at the `[edit system syslog]` hierarchy level. The levels from `emergency` through `info` are in the order from highest severity (greatest effect on functioning) to lowest.

Unlike the other severity levels, the `none` level disables logging of a facility instead of indicating how seriously a triggering event affects routing functions. For more information, see "Disabling the System Logging of a Facility" on page 1285.

Table 126: System Log Message Severity Levels

Value	Severity Level	Description
N/A	`none`	Disables logging of the associated facility to a destination.
0	`emergency`	System panic or other condition that causes the router to stop functioning.
1	`alert`	Conditions that require immediate correction, such as a corrupted system database.
2	`critical`	Critical conditions, such as hard errors.
3	`error`	Error conditions that generally have less serious consequences than errors at the emergency, alert, and critical levels.
4	`warning`	Conditions that warrant monitoring.
5	`notice`	Conditions that are not errors but might warrant special handling.
6	`info`	Events or non-error conditions of interest.
7	`any`	Includes all severity levels.

Default System Log Settings

Table 127 on page 1259 summarizes the default system log settings that apply to all routers that run the Junos OS and specifies which statement to include in the configuration to override the default value.

Table 127: Default System Logging Settings

Setting	Default	Overriding Statement	Instructions

Alternative facility for message forwarded to a remote machine	For change-log: local6
	For conflict-log: local5
	For dfc: local1
	For firewall: local3
	For interactive-commands: local7
	For pfe: local4
		[edit system syslog]
		host *hostname* {
		facility-override *facility*;
		}
			"Changing the Alternative Facility Name for System Log Messages Directed to a Remote Destination" on page 1318

Format of messages logged to a file	Standard Junos OS format, based on UNIX format	[edit system syslog]
		file *filename* {
		structured-data;
		}
			"Logging Messages in Structured-Data Format" on page 1274

Maximum number of files in the archived set	10	[edit system syslog]
		archive {
		files *number*;
		}
		file *filename* {
		archive {
		files *number*;
		}
		}
			"Specifying Log File Size, Number, and Archiving Properties" on page 1274

Maximum size of the log file	M Series, MX Series, and T Series: 1 megabyte (MB)
	TX Matrix: 10 MB
		[edit system syslog]
		archive {
		size *size*;
		}
		file *filename* {
		archive {
		size *size*;
		}
		}
			"Specifying Log File Size, Number, and Archiving Properties" on page 1274

Table 127: Default System Logging Settings *(Continued)*

Setting	Default	Overriding Statement	Instructions
Timestamp format	Month, date, hour, minute, second For example: `Aug 21 12:36:30`	`[edit system syslog]` `time-format` *format*;	"Including the Year or Millisecond in Timestamps" on page 1280

Users who can read log files	`root` user and users with the Junos OS `maintenance` permission	`[edit system syslog]`
		`archive {`
		`  world-readable;`
		`}`
		`file` *filename* `{`
		`  archive {`
		`  world-readable;`
		`  }`
		`}`
			"Specifying Log File Size, Number, and Archiving Properties" on page 1274

Platform-Specific Default System Log Messages

The following messages are generated by default on specific routers. To view any of these types of messages, you must configure at least one destination for messages as described in "Junos OS Minimum System Logging Configuration" on page 1269.

- To log the kernel process message on an M Series, MX Series, or T Series router, include the `kernel info` statement at the appropriate hierarchy level:

```
[edit system syslog]
(console | file filename | host destination | user username) {
    kernel info;
}
```

- On a routing matrix composed of a TX Matrix router and T640 routers, the primary Routing Engine on each T640 router forwards all messages with a severity of `info` and higher to the primary Routing

Engine on the TX Matrix router. This is equivalent to the following *configuration statement* included on the TX Matrix router:

```
[edit system syslog]
host scc-master {
    any info;
}
```

- Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, likewise on a routing matrix composed of a TX Matrix Plus router with connected T1600 or T4000 routers, the primary Routing Engine on each T1600 or T4000 LCC forwards to the primary Routing Engine on the TX Matrix Plus router all messages with a severity of `info` and higher. This is equivalent to the following configuration statement included on the TX Matrix Plus router:

  NOTE: From the perspective of the user interface, the routing matrix appears as a single router. The TX Matrix Plus router controls all the T1600 or T4000 routers connected to it in the routing matrix.

```
[edit system syslog]
host sfc0-master {
    any info;
}
```

Interpret Messages Generated in Standard Format

The syntax of a standard-format message generated by a Junos OS process or subroutine library depends on whether it includes the below priority informations:

- When the `explicit-priority` statement is included at the [*filename*] or [*hostname*] hierarchy level, a system log message has the following syntax:

```
timestamp       message-source: %facility-severity-TAG: message-text
```

- When directed to the console or to users, or when the `explicit-priority` statement is not included for files or remote hosts, a system log message has the following syntax:

```
timestamp        message-source: TAG: message-text
```

Table 128 on page 1262 describes the message fields.

Table 128: Fields in Standard-Format Messages

Field	Description
timestamp	Time at which the message was logged.
message-source	Identifier of the process or component that generates the message and the routing platform on which the message was logged. For Junos OS, this field includes two or more subfields: hostname, process and process ID (PID). For Junos OS Evolved, this field includes a hostname with an appended node name, a process name, and PID. If the `alternate-format` statement is configured at the [edit system syslog] hierarchy level on a Junos OS Evolved device, the node name is not appended to the hostname, but is prepended to the process name instead. The alternate message format for Junos OS Evolved ensures the same hostname format as Junos OS messages. If the process does not report its PID, the PID is not displayed. The message source subfields are displayed in the following format: *hostname process[process-ID]*
facility	Code that specifies the facility to which the system log message belongs. For a mapping of codes to facility names, see Table: Facility Codes Reported in Priority Information in "Including Priority Information in System Log Messages" on page 1276.
severity	Numerical code that represents the severity level assigned to the system log message. For a mapping of codes to severity names, see Table: Numerical Codes for Severity Levels Reported in Priority Information in "Including Priority Information in System Log Messages" on page 1276.
TAG	Text string that uniquely identifies the message, in all uppercase letters and using the underscore (_) to separate words. The tag name begins with a prefix that indicates the generating software process or library. The entries in this reference are ordered alphabetically by this prefix. Not all processes on a routing platform use tags, so this field does not always appear.

Table 128: Fields in Standard-Format Messages *(Continued)*

Field	Description
message-text	Text of the message.

Manage Host OS System Log and Core Files

IN THIS SECTION

- View Log Files On the Host OS System | **1264**
- Copy Log Files From the Host System To the Switch | **1264**
- View Core Files On the Host OS System | **1264**
- Copy Core Files From the Host System To the Switch | **1265**
- Clean Up Temporary Files on the Host OS | **1265**

On Junos OS switches with a host OS, the Junos OS might generates system log messages (also called *syslog messages*) to record events that occur on the switch, including the following:

- Routine operations, such as a user login into the configuration database.
- Failure and error conditions.
- Emergency or critical conditions, such as power-down of the switch due to excessive temperature.

On OCX Series switches:

- System log messages are logged in the **/var/log/dcpfe.log** file in the host OS in the following scenarios:
  - When the forwarding daemon is initialized.
  - Messages are tagged as emergency (LOG_EMERG). A copy of the message is also sent to the **/var/log** directory on the switch.
- Messages from processes are available on the host system in the **/var/log** directory. System log messages from the host chassis management process are recorded in the **lcmd.log** file in the **/var/log** directory.

On QFX switches with a host OS:

- The Junos OS and host OS record log messages for system and process events, and generate core files upon certain system failures.
- These files are stored in directories such as /var/log for log messages, and /var/tmp or /var/crash for core files, depending on the type of host OS running on the switch.

For diagnostic purposes, you can access these host OS system log and core files from the Junos OS CLI on the switch. You can also clean up directories where the host OS stores temporary log and other files.

This topic includes these sections:

View Log Files On the Host OS System

To view a list of the log files created on the host OS, enter the following command:

```
user@switch> show app-engine logs
```

Copy Log Files From the Host System To the Switch

To copy log files from the host OS to the switch, enter the following command:

```
user@switch> request app-engine file-copy log from-jhost source to-vjunos destination
```

For example, to copy the *lcmd* log file to the switch, enter the following command:

```
user@switch> request app-engine file-copy log from-jhost lcmd.log  to-vjunos /var/tmp
```

View Core Files On the Host OS System

To view the list of core files generated and stored on the host OS system, enter the following command:

```
user@switch> show app-engine crash
```

The list might look like this example output:

```
Compute cluster: default-cluster
  Compute node: default-node

     Crash Info
```

```
==========
total 13480
-rw-r--r-- 1 root root  178046 Feb 14 23:08 localhost.lcmd.26653.1455520135.core.tgz
-rw-r--r-- 1 root root 4330343 Feb 15 00:45 localhost.dcpfe.7155.1455525926.core.tgz
-rw-r--r-- 1 root root 4285901 Feb 15 01:49 localhost.dcpfe.25876.1455529782.core.tgz
-rw-r--r-- 1 root root 4288508 Feb 15 02:39 localhost.dcpfe.713.1455532774.core.tgz
-rw-r--r-- 1 root root  264079 Feb 15 17:02 localhost.lcmd.1144.1455584540.core.tgz
```

Copy Core Files From the Host System To the Switch

To copy core files from the host OS to the switch, enter the following command:

```
user@switch> request app-engine file-copy crash from-jhost source to-vjunos destination-dir-or-file-path
```

When the destination Junos OS path is a directory, the source filename is used by default. To rename the file at the destination, enter the destination argument as a full path including the desired filename.

For example, to copy the *localhost.lcmd.26653.1455520135.core.tgz* core archive file to the switch, enter the following command:

```
user@switch> request app-engine file-copy crash from-jhost
localhost.lcmd.26653.1455520135.core.tgz to-vjunos /var/tmp
```

To see the results on the switch, enter the following command:

```
user@switch> show system core-dumps
re0:
------------------------------------------------------------------------
-rw-r--r--  1 root  field     178046 Feb 15 17:15 /var/tmp/
localhost.lcmd.26653.1455520135.core.tgz
total files: 1
```

Clean Up Temporary Files on the Host OS

To remove temporary files created on the host OS, enter the following command:

```
user@switch> request app-engine cleanup
```

For example, the following sample output on a switch with a Linux host OS shows cleanup of temporary files stored in /var/tmp:

```
Compute cluster: default-cluster

  Compute node: default-node

   Cleanup (/var/tmp)
   =======
```

Release History Table

Release	Description
15.1X49-D10	Starting in Junos OS Release 15.1X49-D10 and Junos OS Release 17.3R1, likewise on a routing matrix composed of a TX Matrix Plus router with connected T1600 or T4000 routers, the primary Routing Engine on each T1600 or T4000 LCC forwards to the primary Routing Engine on the TX Matrix Plus router all messages with a severity of `info` and higher.

System Logging on a Single-Chassis System

IN THIS SECTION

- Single-Chassis System Logging Configuration Overview | **1267**
- Junos OS System Log Configuration Statements | **1268**
- Junos OS Minimum System Logging Configuration | **1269**
- Example: Configure System Log Messages | **1270**
- Log Messages in Structured-Data Format | **1274**
- Specify Log File Size, Number, and Archiving Properties | **1274**
- Include Priority Information in System Log Messages | **1276**
- System Log Facility Codes and Numerical Codes Reported in Priority Information | **1278**
- Include the Year or Millisecond in Timestamps | **1280**
- Use Strings and Regular Expressions to Refine the Set of Logged Messages | **1281**

- Junos System Log Regular Expression Operators for the match Statement | **1284**
- Disable the System Logging of a Facility | **1285**
- Examples: Configure System Logging | **1286**
- Examples: Assign an Alternative Facility | **1288**

Single-Chassis System Logging Configuration Overview

The Junos system logging utility is similar to the UNIX `syslogd` utility. This section describes how to configure system logging for a single-chassis system that runs the Junos OS.

System logging configuration for the Junos-FIPS software and for Juniper Networks devices in a Common Criteria environment is the same as for the Junos OS. For more information, see the *Secure Configuration Guide for Common Criteria and Junos-FIPS*.

For information about configuring system logging for a routing matrix composed of a TX Matrix router and T640 routers, see "Configuring System Logging for a TX Matrix Router" on page 1290.

Each system log message belongs to a *facility*, which groups together related messages. Each message is also preassigned a *severity level*, which indicates how seriously the triggering event affects router functions. You always specify the facility and severity of the messages to include in the log. For more information, see "Specifying the Facility and Severity of Messages to Include in the Log" on page 1310.

You direct messages to one or more destinations by including the appropriate statement at the `[edit system syslog]` hierarchy level:

- To a named file in a local file system, by including the `file` statement. See "Directing System Log Messages to a Log File" on page 1313.
- To the terminal session of one or more specific users (or all users) when they are logged in to the router, by including the `user` statement. See "Directing System Log Messages to a User Terminal" on page 1314.
- To the router console, by including the `console` statement. See "Directing System Log Messages to the Console" on page 1315.
- To a remote machine that is running the **syslogd** utility, by including the `host` statement. See *Directing System Log Messages to a Remote Machine*.

By default, messages are logged in a standard format, which is based on a UNIX system log format; for detailed information about message formatting, see the System Log Explorer. You can alter the content and format of logged messages in the following ways:

- You can log messages to a file in structured-data format instead of the standard Junos format. Structured-data format provides more information without adding significant length, and makes it easier for automated applications to extract information from the message. For more information, see "Logging Messages in Structured-Data Format" on page 1274.

- A message's facility and severity level are together referred to as its *priority*. By default, the standard Junos format for messages does not include priority information (structured-data format includes a priority code by default.) To include priority information in standard-format messages directed to a file or a remote destination, include the `explicit-priority` statement. For more information, see "Including Priority Information in System Log Messages" on page 1276.

- By default, the standard Junos format for messages specifies the month, date, hour, minute, and second when the message was logged. You can modify the timestamp on standard-format system log messages to include the year, the millisecond, or both. (Structured-data format specifies the year and millisecond by default.) For more information, see "Including the Year or Millisecond in Timestamps" on page 1280.

- When directing messages to a remote machine, you can specify the IP address that is reported in messages as their source. You can also configure features that make it easier to separate messages generated by Junos OS or messages generated on particular devices. For more information, see *Directing System Log Messages to a Remote Machine*.

- The predefined facilities group together related messages, but you can also use regular expressions to specify more exactly which messages from a facility are logged to a file, a user terminal, or a remote destination. For more information, see "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

NOTE: During a commit check, warnings about the **traceoptions** configuration (for example, mismatch in trace file sizes or number of trace files) are not displayed on the console. However, these warnings are logged in the system log messages when the new configuration is committed.

Junos OS System Log Configuration Statements

To configure the switch to log system messages, include the `syslog` statement at the `[edit system]` hierarchy level:

```
[edit system]
syslog {
    archive <files number> <size size> <world-readable | no-world-readable>;
```

```
    console {
        facility severity;
    }
    file filename {
        facility severity;
        archive <archive-sites (ftp-url <password password>)> <files number> <size size> <start-
time "YYYY-MM-DD.hh:mm"> <transfer-interval minutes> <world-readable | no-world-readable>;
        explicit-priority;
        match "regular-expression";
        structured-data {
            brief;
        }
    }
    host hostname {
        facility severity;
        explicit-priority;
        facility-override facility;
        log-prefix string
        match "regular-expression";
    }
    source-address source-address;
    time-format (year | millisecond | year millisecond);
    user (username | *) {
        facility severity;
        match "regular-expression";
    }
}
```

Junos OS Minimum System Logging Configuration

To record or view system log messages, you must include the `syslog` statement at the `[edit system]` hierarchy level. Specify at least one destination for the messages, as described in Table 129 on page 1270. For more information about the configuration statements, see "Single-Chassis System Logging Configuration Overview" on page 1267.

Table 129: Minimum Configuration Statements for System Logging

Destination	Minimum Configuration Statements

File	[edit system syslog]
	file *filename* {
	    facility severity;
	}

Terminal session of one, several, or all users	[edit system syslog]
	user (*username* | *) {
	    facility severity;
	}

Router or switch console	[edit system syslog]
	console {
	    facility severity;
	}

Remote machine or the other Routing Engine on the router or switch	[edit system syslog]
	host (*hostname* | other-routing-engine) {
	    facility severity;
	}

Example: Configure System Log Messages

IN THIS SECTION

- Requirements | **1271**
- Overview | **1271**
- Configuration | **1271**

The QFabric system monitors events that occur on its component devices and distributes system log messages about those events to all external system log message servers (hosts) that are configured. Component devices may include Node devices, Interconnect devices, Director devices, and the Virtual Chassis. Messages are stored for viewing only in the QFabric system database. To view the messages, issue the `show log` command.

This example describes how to configure system log messages on the QFabric system.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 12.2
- QFabric system
- External servers that can be configured as system log message hosts

Overview

Component devices that generate system log message events may include Node devices, Interconnect devices, Director devices, and the control plane switches. The following configuration example includes these components in the QFabric system:

- Director software running on the Director group
- Control plane switches
- Interconnect device
- Multiple Node devices

Configuration

IN THIS SECTION

- Procedure | 1272

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the `[edit]` hierarchy level.

```
set system syslog host 10.1.1.12 any error
set system syslog file qflogs
set system syslog file qflogs structured-data brief
set system syslog file qflogs archive size 1g
```

Step-by-Step Procedure

The following example requires that you navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure system messages from the QFabric Director device:

1. Specify a host, any facility, and the `error` severity level.

```
[edit system syslog]
user@switch# set host 10.1.1.12 any error
```

> **NOTE**: You can configure more than one system log message server (host). The QFabric system sends the messages to each server configured.

2. (Optional) Specify a filename to capture log messages.

> **NOTE**: On the QFabric system, a syslog file named **messages** is configured implicitly with facility and severity levels of `any any` and a file size of 100 MBs. Therefore, you cannot specify

the filename **messages** in your configuration, and automatic command completion does not work for that filename.

```
[edit system syslog]
user@switch# set file qflogs structured-data brief
user@switch# set file qflogs
```

3. (Optional) Configure the maximum size of your system log message archive file. This example specifies an archive size of 1 GB.

```
[edit system syslog]
user@switch# set file qflogs archive size 1g
```

Results

From configuration mode, confirm your configuration by entering the `show system` command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
[edit]
user@switch# show system
syslog {
    file qflogs {
    }
    host 10.1.1.12 {
        any error;
    }
}
```

If you are done configuring the device, enter `commit` from configuration mode.

SEE ALSO

syslog (QFabric System)

show log

Log Messages in Structured-Data Format

You can log messages to a file in structured-data format instead of the standard Junos OS format. Structured-data format provides more information without adding significant length, and makes it easier for automated applications to extract information from a message.

The structured-data format complies with Internet standard RFC 5424, *The Syslog Protocol*, which is at https://tools.ietf.org/html/rfc5424. The RFC establishes a standard message format regardless of the source or transport protocol for logged messages.

To output messages to a file in structured-data format, include the `structured-data` statement at the `[edit system syslog file filename]` hierarchy level:

```
[edit system syslog file filename]
facility severity;
structured-data {
    brief;
}
```

The optional `brief` statement suppresses the English-language text that appears by default at the end of a message to describe the error or event.

The structured format is used for all messages logged to the file that are generated by a Junos process or software library.

NOTE: If you include either or both of the `explicit-priority` and `time-format` statements along with the `structured-data` statement, they are ignored. These statements apply to the standard Junos OS system log format, not to structured-data format.

Specify Log File Size, Number, and Archiving Properties

To prevent log files from growing too large, by default the Junos OS system logging utility writes messages to a sequence of files of a defined size. The files in the sequence are referred to as *archive* files to distinguish them from the *active* file to which messages are currently being written. The default maximum size depends on the platform type:

- 128 kilobytes (KB) for EX Series switches
- 1 megabyte (MB) for M Series, MX Series, and T Series routers

- 10 MB for TX Matrix or TX Matrix Plus routers
- 1 MB for the QFX Series

When an active log file called ***logfile*** reaches the maximum size, the logging utility closes the file, compresses it, and names the compressed archive file ***logfile*.0.gz**. The logging utility then opens and writes to a new active file called ***logfile***. This process is also known as file rotation. When the new ***logfile*** reaches the configured maximum size, ***logfile*.0.gz** is renamed ***logfile*.1.gz**, and the new ***logfile*** is closed, compressed, and renamed ***logfile*.0.gz**. By default, the logging utility creates up to 10 archive files in this manner. When the maximum number of archive files is reached and when the size of the active file reaches the configured maximum size, the contents of the last archived file are overwritten by the current active file. The logging utility by default also limits the users who can read log files to the `root` user and users who have Junos OS `maintenance` permission.

Junos OS provides a configuration statement `log-rotate-frequency` that configures the system log file rotation frequency by configuring the time interval for checking the log file size. The frequency can be set to a value of 1 minute through 59 minutes. The default frequency is 15 minutes.

To configure the log rotation frequency, include the `log-rotate-frequency` statement at the `[edit system syslog]` hierarchy level.

You can include the `archive` statement to change the maximum size of each file, how many archive files are created, and who can read log files.

To configure values that apply to all log files, include the `archive` statement at the `[edit system syslog]` hierarchy level:

```
archive  <files number> <size size> <world-readable | no-world-readable>;
```

To configure values that apply to a specific log file, include the `archive` statement at the `[edit system syslog file filename]` hierarchy level:

```
archive <archive-sites (ftp-url  <password password>)> <files number> <size size> <start-
time "YYYY-MM-DD.hh:mm"> <transfer-interval minutes> <world-readable | no-world-readable> ;
```

`archive-sites` *`site-name`* specifies a list of archive sites that you want to use for storing files. The *`site-name`* value is any valid FTP URL to a destination. If more than one site name is configured, a list of archive sites for the system log files is created. When a file is archived, the router or switch attempts to transfer the file to the first URL in the list, moving to the next site only if the transfer does not succeed. The log file is stored at the archive site with the specified log filename. For information about how to specify valid FTP URLs, see Format for Specifying Filenames and URLs in Junos OS CLI Commands.

`binary-data` Mark file as containing binary data. This allows proper archiving of binary files, such as WTMP files (login records for UNIX based systems). To restore the default setting, include the `no-binary-data` statement.

`files` *`number`* specifies the number of files to create before the oldest file is overwritten. The value can be from 1 through 1000.

`size` *`size`* specifies the maximum size of each file. The value can be from 64 KB (64k) through 1 gigabyte (1g); to represent megabytes, use the letter `m` after the integer. There is no space between the digits and the `k`, `m`, or `g` units letter.

`start-time "`*`YYYY-MM-DD.hh:mm`*`"` defines the date and time in the local time zone for a one-time transfer of the active log file to the first reachable site in the list of sites specified by the `archive-sites` statement.

`transfer-interval` *`interval`* defines the amount of time the current log file remains open (even if it has not reached the maximum possible size) and receives new statistics before it is closed and transferred to an archive site. This interval value can be from 5 through 2880 minutes.

`world-readable` enables all users to read log files. To restore the default permissions, include the `no-world-readable` statement.

Include Priority Information in System Log Messages

The facility and severity level of a message are together referred to as its *priority*. By default, messages logged in the standard Junos OS format do not include information about priority. To include priority information in standard-format messages directed to a file, include the `explicit-priority` statement at the `[edit system syslog file` *`filename`*`]` hierarchy level:

```
[edit system syslog file filename]
facility severity;
explicit-priority;
```

NOTE: Messages logged in structured-data format include priority information by default. If you include the `structured-data` statement at the `[edit system syslog file` *`filename`*`]` hierarchy level along with the `explicit-priority` statement, the `explicit-priority` statement is ignored and messages are logged in structured-data format.

For information about the `structured-data` statement, see "Logging Messages in Structured-Data Format" on page 1274.

To include priority information in messages directed to a remote machine or the other Routing Engine, include the `explicit-priority` statement at the `[edit system syslog host (hostname | other-routing-engine)]` hierarchy level:

```
[edit system syslog host (hostname | other-routing-engine)]
facility severity;
explicit-priority;
```

NOTE: The `other-routing-engine` option does not apply to the QFX Series.

The priority recorded in a message always indicates the original, local facility name. If the `facility-override` statement is included for messages directed to a remote destination, the Junos OS system logging utility still uses the alternative facility name for the messages themselves when directing them to the remote destination. For more information, see "Changing the Alternative Facility Name for System Log Messages Directed to a Remote Destination" on page 1318.

When the `explicit-priority` statement is included, the Junos OS logging utility prepends codes for the facility name and severity level to the message tag name, if the message has one:

```
FACILITY-severity[-TAG]
```

(The tag is a unique identifier assigned to some Junos OS system log messages.)

In the following example, the `CHASSISD_PARSE_COMPLETE` message belongs to the `daemon` facility and is assigned severity `info` (6):

```
Aug 21 12:36:30 router1 chassisd[522]: %DAEMON-6-CHASSISD_PARSE_COMPLETE: Using new configuration
```

When the `explicit-priority` statement is not included, the priority does not appear in the message:

```
Aug 21 12:36:30 router1 chassisd[522]: CHASSISD_PARSE_COMPLETE: Using new configuration
```

System Log Facility Codes and Numerical Codes Reported in Priority Information

Table 130 on page 1278 lists the facility codes that can appear in system log messages and maps them to facility names.

> **NOTE**: If the second column in Table 130 on page 1278 does not include the Junos OS facility name for a code, the facility cannot be included in a statement at the `[edit system syslog]` hierarchy level. Junos OS might use the facilities in Table 130 on page 1278—and others that are not listed—when reporting on internal operations.

Table 130: Facility Codes Reported in Priority Information

Code	Junos Facility Name	Type of Event or Error
AUTH	authorization	Authentication and authorization attempts
AUTHPRIV		Authentication and authorization attempts that can be viewed by superusers only
CHANGE	change-log	Changes to Junos OS configuration
CONFLICT	conflict-log	Specified configuration is invalid on the router type
CONSOLE		Messages written to **/dev/console** by the kernel console output r
CRON		Actions performed or errors encountered by the cron process
DAEMON	daemon	Actions performed or errors encountered by system processes
DFC	dfc	Actions performed or errors encountered by the dynamic flow capture process
FIREWALL	firewall	Packet filtering actions performed by a firewall filter

Table 130: Facility Codes Reported in Priority Information *(Continued)*

Code	Junos Facility Name	Type of Event or Error
FTP	ftp	Actions performed or errors encountered by the FTP process
INTERACT	interactive-commands	Commands issued at the Junos OS CLI prompt or invoked by a client application such as a Junos XML protocol or NETCONF client
KERN	kernel	Actions performed or errors encountered by the Junos kernel
NTP		Actions performed or errors encountered by the Network Time Protocol (NTP)
PFE	pfe	Actions performed or errors encountered by the Packet Forwarding Engine
SYSLOG		Actions performed or errors encountered by the Junos system logging utility
USER	user	Actions performed or errors encountered by user-space processes

Table 131 on page 1279 lists the numerical severity codes that can appear in system log messages and maps them to severity levels.

Table 131: Numerical Codes for Severity Levels Reported in Priority Information

Numerical Code	Severity Level	Description
0	emergency	System panic or other condition that causes the ro functioning
1	alert	Conditions that require immediate correction, such corrupted system database
2	critical	Critical conditions, such as hard errors

Table 131: Numerical Codes for Severity Levels Reported in Priority Information *(Continued)*

Numerical Code	Severity Level	Description
3	error	Error conditions that generally have less serious co than errors in the emergency, alert, and critical leve
4	warning	Conditions that warrant monitoring
5	notice	Conditions that are not errors but might warrant sp
6	info	Events or nonerror conditions of interest
7	debug	Software debugging messages (these appear only if support representative has instructed you to config severity level)

Include the Year or Millisecond in Timestamps

By default, the timestamp recorded in a standard-format system log message specifies the month, date, hour, minute, and second when the message was logged, as in the following example:

```
Aug 21 12:36:30
```

To include the year, the millisecond, or both in the timestamp, include the `time-format` statement at the `[edit system syslog]` or `[edit security log]` hierarchy levels:

```
[edit system syslog]
time-format (year | millisecond | year millisecond);
```

However, the timestamp for traceoption messages is specified in milliseconds by default, and is independent of the `[edit system syslog time-format]` statement.

The modified timestamp is used in messages directed to each destination configured by a `file`, `console`, or `user` statement at the `[edit system syslog]` hierarchy level, but not to destinations configured by a `host` statement.

NOTE: By default, in a FreeBSD console, the additional time information is not available in system log messages directed to each destination configured by a `host` statement. However, in a Junos OS specific implementation using the FreeBSD console, the additional time information is available in system log messages directed to each destination.

The following example illustrates the format for a timestamp that includes both the millisecond (401) and the year (2006):

```
Aug 21 12:36:30.401 2006
```

NOTE: Messages logged in structured-data format include the year and millisecond by default. If you include the structured-data statement at the `[edit system syslog file filename]` hierarchy level along with the `time-format` statement, the `time-format` statement is ignored and messages are logged in structured-data format.

For information about the `structured-data` statement, see "Logging Messages in Structured-Data Format" on page 1274.

Use Strings and Regular Expressions to Refine the Set of Logged Messages

The predefined facilities group together related messages, but you can also match messages against strings and regular expressions to refine which messages from a facility are logged to a file, a user terminal, or a remote destination.

The `match-strings` and `match` configuration statements enable you to match system log messages against a string or regular expression, respectively. You can include these statements at the following hierarchy levels:

- `[edit system syslog file filename]` (for a file)
- `[edit system syslog user (username | *)]` (for a specific user session or for all user sessions on a terminal)
- `[edit system syslog host (hostname | other-routing-engine)]` (for a remote destination)

To evaluate messages against a regular expression and only log matching messages to the given destination, include the match statement and specify the regular expression:

```
match "regular-expression";
```

Starting with Junos OS Release 16.1, you can use simple string comparisons to more efficiently filter messages, because it is less CPU-intensive than matching against complex regular expressions. To specify the text string that must appear in a message for the message to be logged to a destination, include the match-strings statement and specify the matching string or list of strings:

```
match-strings string-name;
```

```
match-strings [string1 string2];
```

The match-strings and match statements select messages with the configured facility and severity that match the given string or regular expression. The match-strings statement performs a simple string comparison, and as a result, it is less CPU-intensive than using the match statement to match against complex regular expressions. If you configure both the match and match-strings statements for the same destination, Junos OS evaluates the match-strings condition first; if the message includes any of the configured substrings, then the message is logged and the match condition is not evaluated. If the match-strings condition is not satisfied, then the system evaluates the message against the regular expression in the match configuration statement.

When specifying regular expressions for the match statement, use the notation defined in POSIX Standard 1003.2 for extended (modern) UNIX regular expressions. Explaining regular expression syntax is beyond the scope of this document, but POSIX standards are available from the Institute of Electrical and Electronics Engineers (IEEE, http://www.ieee.org).

Table 132 on page 1283 specifies which character or characters are matched by some of the regular expression operators that you can use in the match statement. In the descriptions, the term term refers to either a single alphanumeric character or a set of characters enclosed in square brackets, parentheses, or braces.

NOTE: The match statement is not case-sensitive.

Table 132: Regular Expression Operators for the match Statement

Operator	Matches
. (period)	One instance of any character except the space.
* (asterisk)	Zero or more instances of the immediately preceding term.
+ (plus sign)	One or more instances of the immediately preceding term.
? (question mark)	Zero or one instance of the immediately preceding term.
| (pipe)	One of the terms that appears on either side of the pipe operator.
! (exclamation point)	Any string except the one specified by the expression, when the exclamation point appears at the start of the expression. Use of the exclamation point is Junos OS-specific.
^ (caret)	Start of a line, when the caret appears outside square brackets. One instance of any character that does not follow it within square brackets, when the caret is the first character inside square brackets.
$ (dollar sign)	End of a line.
[ ] (paired square brackets)	One instance of one of the enclosed alphanumeric characters. To indicate a range of characters, use a hyphen (-) to separate the beginning and ending characters of the range. For example, [a-z0-9] matches any letter or number.
( ) (paired parentheses)	One instance of the evaluated value of the enclosed term. Parentheses are used to indicate the order of evaluation in the regular expression.

Using Strings and Regular Expressions

Filter messages that belong to the `interactive-commands` facility, directing those that include the string `configure` to the terminal of the root user:

```
[edit system syslog]
user root {
```

```
    interactive-commands any;
    match-strings configure;
}
```

Messages like the following appear on the `root` user's terminal when a user issues a `configure` command to enter configuration mode:

```
timestamp router-name mgd[PID]: UI_CMDLINE_READ_LINE: User 'user', command 'configure private'
```

Filter messages that belong to the `daemon` facility and have a severity of `error` or higher, directing them to the file **/var/log/process-errors**. Omit messages generated by the SNMP process (snmpd), instead directing them to the file **/var/log/snmpd-errors**:

```
[edit system syslog]
file process-errors {
    daemon error;
    match "!(.*snmpd.*)";
}
file snmpd-errors {
    daemon error;
    match-strings snmpd;
}
```

Junos System Log Regular Expression Operators for the match Statement

Table 133: Regular Expression Operators for the match Statement

Operator	Matches
. (period)	One instance of any character except the space.
* (asterisk)	Zero or more instances of the immediately preceding term.
+ (plus sign)	One or more instances of the immediately preceding term.

Table 133: Regular Expression Operators for the match Statement *(Continued)*

Operator	Matches
? (question mark)	Zero or one instance of the immediately preceding term.
| (pipe)	One of the terms that appear on either side of the pipe operator.
! (exclamation point)	Any string except the one specified by the expression, when the exclamation point appears at the start of the expression. Use of t exclamation point is Junos OS–specific.
^ (caret)	The start of a line, when the caret appears outside square bracket One instance of any character that does not follow it within squa brackets, when the caret is the first character inside square brack
$ (dollar sign)	The end of a line.
[ ] (paired square brackets)	One instance of one of the enclosed alphanumeric characters. To indicate a range of characters, use a hyphen (-) to separate the beginning and ending characters of the range. For example, `[a-z0` matches any letter or number.
( ) (paired parentheses)	One instance of the evaluated value of the enclosed term. Parentheses are used to indicate the order of evaluation in the re expression.

Disable the System Logging of a Facility

To disable the logging of messages that belong to a particular facility, include the *facility* `none` statement in the configuration. This statement is useful when, for example, you want to log messages that have the same severity level and belong to all but a few facilities. Instead of including a statement for each facility you want to log, you can include the `any` *severity* statement and then a *facility* `none` statement for each facility that you do not want to log. For example, the following logs all messages at the `error` level or

higher to the console, except for messages from the `daemon` and `kernel` facilities. Messages from those facilities are logged to the file **>/var/log/internals** instead:

```
[edit system syslog]
console {
    any error;
    daemon none;
    kernel none;
}
file internals {
    daemon info;
    kernel info;
}
```

Examples: Configure System Logging

The following example shows how to configure the logging of messages about all commands entered by users at the CLI prompt or invoked by client applications such as Junos OS XML protocol or NETCONF client applications, and all authentication or authorization attempts, both to the file **cli-commands** and to the terminal of any user who is logged in:

```
[edit system]
syslog {
    file cli-commands {
        interactive-commands info;
        authorization info;
    }
    user * {
        interactive-commands info;
        authorization info;
    }
}
```

The following example shows how to configure the logging of all changes in the state of alarms to the file **/var/log/alarms**:

```
[edit system]
syslog {
```

```
    file alarms {
        kernel warning;
    }
}
```

The following example shows how to configure the handling of messages of various types, as described in the comments. Information is logged to two files, to the terminal of user `alex`, to a remote machine, and to the console:

```
[edit system]
syslog {
    /* write all security-related messages to file /var/log/security */
    file security {
        authorization info;
        interactive-commands info;
    }
    /* write messages about potential problems to file /var/log/messages: */
    /* messages from "authorization" facility at level "notice" and above, */
    /* messages from all other facilities at level "warning" and above */
    file messages {
        authorization notice;
        any warning;
    }
    /* write all messages at level "critical" and above to terminal of user "alex" if */
    /* that user is logged in */
    user alex {
        any critical;
    }
    /* write all messages from the "daemon" facility at level "info" and above, and */
    /* messages from all other facilities at level "warning" and above, to the */
    /* machine monitor.mycompany.com */
    host monitor.mycompany.com {
        daemon info;
        any warning;
    }
    /* write all messages at level "error" and above to the system console */
    console {
        any error;
    }
}
```

The following example shows how to configure the handling of messages generated when users issue Junos OS CLI commands, by specifying the `interactive-commands` facility at the following severity levels:

- `info`—Logs a message when users issue any command at the CLI operational or configuration mode prompt. The example writes the messages to the file **/var/log/user-actions**.
- `notice`—Logs a message when users issue the configuration mode commands `rollback` and `commit`. The example writes the messages to the terminal of user `philip`.
- `warning`—Logs a message when users issue a command that restarts a software process. The example writes the messages to the console.

```
[edit system]
syslog {
    file user-actions {
        interactive-commands info;
    }
    user philip {
        interactive-commands notice;
    }
    console {
        interactive-commands warning;
    }
}
```

Examples: Assign an Alternative Facility

Log all messages generated on the local routing platform at the error level or higher to the `local0` facility on the remote machine called `monitor.mycompany.com`:

```
[edit system syslog]
host monitor.mycompany.com {
    any error;
    facility-override local0;
}
```

Configure routing platforms located in California and routing platforms located in New York to send messages to a single remote machine called `central-logger.mycompany.com`. The messages from California are

assigned alternative facility `local0` and the messages from New York are assigned to alternative facility `local2`.

- Configure California routing platforms to aggregate messages in the `local0` facility:

```
[edit system syslog]
host central-logger.mycompany.com {
    change-log info;
    facility-override local0;
}
```

- Configure New York routing platforms to aggregate messages in the `local2` facility:

```
[edit system syslog]
host central-logger.mycompany.com {
    change-log info;
    facility-override local2;
}
```

On `central-logger` you can then configure the system logging utility to write messages from the `local0` facility to the file `california-config` and the messages from the `local2` facility to the file `new-york-config`.

System Logging for a TX Matrix or TX Matrix Plus Router

IN THIS SECTION

- Configuring System Logging for a TX Matrix Router | **1290**
- Configuring System Logging for a TX Matrix Plus Router | **1292**
- Configuring Message Forwarding to the TX Matrix Router | **1294**
- Configuring Message Forwarding to the TX Matrix Plus Router | **1295**
- Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Router | **1296**

- Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Plus Router | **1299**
- Configuring Optional Features for Forwarded Messages on a TX Matrix Router | **1302**
- Configuring Optional Features for Forwarded Messages on a TX Matrix Plus Router | **1304**
- Configuring System Logging Differently on Each T640 Router in a Routing Matrix | **1306**
- Configuring System Logging Differently on Each T1600 or T4000 Router in a Routing Matrix | **1308**

Configuring System Logging for a TX Matrix Router

To configure system logging for all routers in a routing matrix composed of a TX Matrix router and T640 routers, include the `syslog` statement at the `[edit system]` hierarchy level on the TX Matrix router. The `syslog` statement applies to every router in the routing matrix.

```
[edit system]
syslog {
    archive <files number> <size size <world-readable | no-world-readable>;
    console {
        facility severity;
    }
    file filename {
        facility severity;
        archive <archive-sites {ftp-url  <password password>}> <files number> <size size> <start-
time "YYYY-MM-DD.hh:mm"> <transfer-interval minutes> <world-readable | no-world-readable>;
        explicit-priority;
        match "regular-expression";
        structured-data {
            brief;
        }
    }
    host (hostname | other-routing-engine | scc-master) {
        facility severity;
        explicit-priority;
        facility-override facility;
        log-prefix string;
        match "regular-expression";
        source-address source-address;
        port port number;
```

```
    }
    source-address source-address;
    time-format (year | millisecond | year millisecond);
     (username | *) {
        facility severity;
        match "regular-expression";
    }
}
```

When included in the configuration on the TX Matrix router, the following configuration statements have the same effect as on a single-chassis system, except that they apply to every router in the routing matrix:

- `archive`—Sets the size and number of log files on each platform in the routing matrix. See "Specifying Log File Size, Number, and Archiving Properties" on page 1274.
- `console`—Directs the specified messages to the console of each platform in the routing matrix. See "Directing System Log Messages to the Console" on page 1315.
- `file`—Directs the specified messages to a file of the same name on each platform in the routing matrix. See "Directing System Log Messages to a Log File" on page 1313.
- `match`—Limits the set of messages logged to a destination to those that contain (or do not contain) a text string matching a regular expression. See "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

  The separate `match` statement at the `[edit system syslog host scc-master]` hierarchy level applies to messages forwarded from the T640 routers to the TX Matrix router. See "Configuring Optional Features for Forwarded Messages on a TX Matrix Router" on page 1302.
- `port`—Specifies the port number of the remote syslog server.
- `source-address`—Sets the IP address of the router to report in system log messages as the message source, when the messages are directed to the remote machines specified in all *hostname* statements at the `[edit system syslog]` hierarchy level, for each platform in the routing matrix. On a routing matrix composed of a TX Matrix router and T640 routers, the address is not reported by the T640 routers in messages directed to the other Routing Engine on each router or to the TX Matrix router. See "Specifying an Alternative Source Address for System Log Messages Directed to a Remote Destination" on page 1316.
- `structured-data`—Writes messages to a file in structured-data format. See "Logging Messages in Structured-Data Format" on page 1274.
- `time-format`—Adds the millisecond, year, or both to the timestamp in each standard-format message. See "Including the Year or Millisecond in Timestamps" on page 1280.

- user—Directs the specified messages to the terminal session of one or more specified users on each platform in the routing matrix that they are logged in to. See "Directing System Log Messages to a User Terminal" on page 1314.

The effect of the other statements differs somewhat for a routing matrix than for a single-chassis system.

Configuring System Logging for a TX Matrix Plus Router

From the perspective of the user interface, the routing matrix appears as a single router. The TX Matrix Plus router (also called the switch-fabric chassis SFC) controls all the T1600 or T4000 routers (also called the line-card chassis LCC) in the routing matrix.

To configure system logging for all routers in a routing matrix composed of a TX Matrix Plus router with connected T1600 or T4000 LCCs, include the syslog statement at the [edit system] hierarchy level on the SFC. The syslog statement applies to every router in the routing matrix.

```
[edit system]
syslog {
    archive <files number> <size size <world-readable | no-world-readable>;
    console {
        facility severity;
    }
    file filename {
        facility severity;
        archive <archive-sites {ftp-url  <password password>}> <files number> <size size> <start-
time "YYYY-MM-DD.hh:mm"> <transfer-interval minutes> <world-readable | no-world-readable>;
        explicit-priority;
        match "regular-expression";
        structured-data {
            brief;
        }
    }
    host (hostname | other-routing-engine | sfc0-master) {
        facility severity;
        explicit-priority;
        facility-override facility;
        log-prefix string;
        match "regular-expression";
    }
    source-address source-address;
```

```
    time-format (year | millisecond | year millisecond);
     (username | *) {
        facility severity;
        match "regular-expression";
    }
}
```

When included in the configuration on the TX Matrix Plus router, the following configuration statements have the same effect as on a single-chassis system, except that they apply to every router in the routing matrix.

- `archive`—Sets the size and number of log files on each router in the routing matrix. See "Specifying Log File Size, Number, and Archiving Properties" on page 1274.
- `console`—Directs the specified messages to the console of each router in the routing matrix. See "Directing System Log Messages to the Console" on page 1315.
- `file`—Directs the specified messages to a file of the same name on each router in the routing matrix. See "Directing System Log Messages to a Log File" on page 1313.
- `match`—Limits the set of messages logged to a destination to those that contain (or do not contain) a text string matching a regular expression. See "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

  The separate `match` statement at the `[edit system syslog host sfc0-master]` hierarchy level applies to messages forwarded from the T1600 or T4000 LCCs to the SFC. See "Configuring Optional Features for Forwarded Messages on a TX Matrix Plus Router" on page 1304.
- `source-address`—Sets the IP address of the router as the message source in system log messages when the messages are directed to the remote machines specified in all `host` *`hostname`* statements at the `[edit system syslog]` hierarchy level, for each router in the routing matrix. On a routing matrix composed of a TX Matrix Plus router with connected T1600 or T4000 LCCs, the address is not reported by the T1600 or T4000 routers in messages directed to the other Routing Engine on each router or to the TX Matrix Plus router. See "Specifying an Alternative Source Address for System Log Messages Directed to a Remote Destination" on page 1316.
- `structured-data`—Writes messages to a file in structured-data format. See "Logging Messages in Structured-Data Format" on page 1274.
- `time-format`—Adds the millisecond, year, or both to the timestamp in each standard-format message. See "Including the Year or Millisecond in Timestamps" on page 1280.
- `user`—Directs the specified messages to the terminal session of one or more specified users on each router in the routing matrix that they are logged in to. See "Directing System Log Messages to a User Terminal" on page 1314.

The effect of the other statements differs somewhat for a routing matrix than for a single-chassis system.

Configuring Message Forwarding to the TX Matrix Router

By default, the primary Routing Engine on each T640 router forwards to the primary Routing Engine on the TX Matrix router all messages from all facilities with severity level of `info` and higher. To change the facility, the severity level, or both, include the `host scc-master` statement at the `[edit system syslog]` hierarchy level on the TX Matrix router:

```
[edit system syslog]
host scc-master {
    facility severity;
}
```

To disable message forwarding, set the facility to `any` and the severity level to `none`:

```
[edit system syslog]
host scc-master {
    any none;
}
```

In either case, the setting applies to all T640 routers in the routing matrix.

To capture the messages forwarded by the T640 routers (as well as messages generated on the TX Matrix router itself), you must also configure system logging on the TX Matrix router. Direct the messages to one or more destinations by including the appropriate statements at the `[edit system syslog]` hierarchy level on the TX Matrix router:

- To a file, as described in "Directing System Log Messages to a Log File" on page 1313.
- To the terminal session of one or more specific users (or all users), as described in "Directing System Log Messages to a User Terminal" on page 1314.
- To the console, as described in "Directing System Log Messages to the Console" on page 1315.
- To a remote machine that is running the syslogd utility or to the other Routing Engine. For more information, see "Directing Messages to a Remote Destination from the Routing Matrix Based on the TX Matrix Router" on page 1323.

As previously noted, the configuration statements included on the TX Matrix router also configure the same destinations on each T640 router in the routing matrix.

When specifying the severity level for local messages (at the `[edit system syslog (file | host | console | user)]` hierarchy level) and forwarded messages (at the `[edit system syslog host scc-master]` hierarchy level), you can set the same severity level for both, set a lower severity level for local messages, or set a higher severity level for local messages. The following examples describe the consequence of each configuration. (For simplicity, the examples use the `any` facility in every case. You can also specify different severities for different facilities, with more complex consequences.)

Configuring Message Forwarding to the TX Matrix Plus Router

From the perspective of the user interface, the routing matrix appears as a single router. The TX Matrix Plus router (also called the switch-fabric chassis SFC) controls all the T1600 or T4000 routers (also called the line-card chassis LCC) in the routing matrix.

By default, the primary Routing Engine on each connected T1600 or T4000 LCC forwards to the primary Routing Engine on the SFC all messages from all facilities with severity level of `info` and higher. To change the facility, the severity level, or both, include the `host sfc0-master` statement at the `[edit system syslog]` hierarchy level on the SFC:

```
[edit system syslog]
host sfc0-master {
    facility severity;
}
```

To disable message forwarding, set the facility to `any` and the severity level to `none`:

```
[edit system syslog]
host sfc0-master {
    any none;
}
```

In either case, the setting applies to all connected LCCs in the routing matrix.

To capture the messages forwarded by the T1600 or T4000 LCCs (as well as messages generated on the SFC itself), you must also configure system logging on the SFC. Direct the messages to one or more destinations by including the appropriate statements at the `[edit system syslog]` hierarchy level on the SFC:

- To a file, as described in "Directing System Log Messages to a Log File" on page 1313.

- To the terminal session of one or more specific users (or all users), as described in "Directing System Log Messages to a User Terminal" on page 1314.
- To the console, as described in "Directing System Log Messages to the Console" on page 1315.
- To a remote machine that is running the syslogd utility or to the other Routing Engine. For more information, see "Directing Messages to a Remote Destination from the Routing Matrix Based on a TX Matrix Plus Router" on page 1324.

As previously noted, the configuration statements included on the SFC also configure the same destinations on each connected LCC.

When specifying the severity level for local messages (at the `[edit system syslog (file | host | console | user)]` hierarchy level) and forwarded messages (at the `[edit system syslog host sfc0-master]` hierarchy level), you can set the same severity level for both, set a lower severity level for local messages, or set a higher severity level for local messages. The following examples describe the consequence of each configuration. (For simplicity, the examples use the `any` facility in every case. You can also specify different severities for different facilities, with more complex consequences.)

Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Router

IN THIS SECTION

- Messages Logged When the Local and Forwarded Severity Levels Are the Same | **1297**
- Messages Logged When the Local Severity Level Is Lower | **1297**
- Messages Logged When the Local Severity Level Is Higher | **1298**

This topic describes the impact of different local and forwarded severity levels configured for system log messages on a TX Matrix router:

Messages Logged When the Local and Forwarded Severity Levels Are the Same

When the severity level is the same for local and forwarded messages, the log on the TX Matrix router contains all messages from the logs on the T640 routers. For example, you can specify severity `info` for the **/var/log/messages** file, which is the default severity level for messages forwarded by T640 routers:

```
[edit system syslog]
file messages {
    any info;
}
```

Table 134 on page 1297 specifies which messages are included in the logs on the T640 routers and the TX Matrix router.

Table 134: Example: Local and Forwarded Severity Level Are Both info

Log Location	Source of Messages	Lowest Severity Included
T640 router	Local	`info`
TX Matrix router	Local	`info`
	Forwarded from T640 routers	`info`

Messages Logged When the Local Severity Level Is Lower

When the severity level is lower for local messages than for forwarded messages, the log on the TX Matrix router includes fewer forwarded messages than when the severities are the same. Locally generated messages are still logged at the lower severity level, so their number in each log is the same as when the severities are the same.

For example, on a TX Matrix router, you can specify severity `notice` for the **/var/log/messages** file and severity `critical` for forwarded messages:

```
[edit system syslog]
file messages {
    any notice;
}
host scc-master {
```

```
    any critical;
}
```

Table 135 on page 1298 specifies which messages in a routing matrix are included in the logs on the T640 routers and the TX Matrix router. The T640 routers forward only those messages with severity `critical` and higher, so the log on the TX Matrix router does not include the messages with severity `error`, `warning`, or `notice` that the T640 routers log locally.

Table 135: Example: Local Severity Is notice, Forwarded Severity Is critical

Log Location	Source of Messages	Lowest Severity Included
T640 router	Local	`notice`
TX Matrix router	Local	`notice`
	Forwarded from T640 routers	`critical`

Messages Logged When the Local Severity Level Is Higher

When the severity level is higher for local messages than for forwarded messages, the log on the TX Matrix router includes fewer forwarded messages than when the severities are the same, and all local logs contain fewer messages overall.

For example, you can specify severity `critical` for the **/var/log/messages** file and severity `notice` for forwarded messages:

```
[edit system syslog]
file messages {
    any critical;
}
host scc-master {
    any notice;
}
```

Table 136 on page 1299 specifies which messages are included in the logs on the T640 routers and the TX Matrix router. Although the T640 routers forward messages with severity `notice` and higher, the TX Matrix router discards any of those messages with severity lower than `critical` (does not log forwarded

messages with severity `error`, `warning`, or `notice`). None of the logs include messages with severity `error` or lower.

Table 136: Example: Local Severity Is critical, Forwarded Severity Is notice

Log Location	Source of Messages	Lowest Severity Included
T640 router	Local	`critical`
TX Matrix router	Local	`critical`
	Forwarded from T640 routers	`critical`

Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Plus Router

IN THIS SECTION

- Messages Logged When the Local and Forwarded Severity Levels Are the Same | **1299**
- Messages Logged When the Local Severity Level Is Lower | **1300**
- Messages Logged When the Local Severity Level Is Higher | **1301**

This topic describes the impact of different local and forwarded severity levels configured for the system log messages on a TX Matrix Plus router:

Messages Logged When the Local and Forwarded Severity Levels Are the Same

When the severity level is the same for local and forwarded messages, the log on the TX Matrix Plus router contains all messages from the logs on the T1600 routers in the routing matrix. For example, you

can specify severity `info` for the **/var/log/messages** file, which is the default severity level for messages forwarded by T1600 routers:

```
[edit system syslog]
file messages {
    any info;
}
```

Table 137 on page 1300 specifies which messages in a routing matrix based on a TX Matrix Plus router are included in the logs on the T1600 routers and the TX Matrix Plus router:

Table 137: Example: Local and Forwarded Severity Level Are Both info

Log Location	Source of Messages	Lowest Severity Included
T1600 router	Local	`info`
TX Matrix Plus router	Local	`info`
	Forwarded from T1600 routers	`info`

Messages Logged When the Local Severity Level Is Lower

When the severity level is lower for local messages than for forwarded messages, the log on the TX Matrix Plus router includes fewer forwarded messages than when the severities are the same. Locally generated messages are still logged at the lower severity level, so their number in each log is the same as when the severities are the same.

For example, on a TX Matrix Plus router, you can specify severity `notice` for the **/var/log/messages** file and severity `critical` for forwarded messages:

```
[edit system syslog]
file messages {
    any notice;
}
host sfc0-master {
    any critical;
}
```

Table 138 on page 1301 specifies which messages in a routing matrix are included in the logs on the T1600 routers and the TX Matrix Plus router. The T1600 routers forward only those messages with severity `critical` and higher, so the log on the TX Matrix Plus router does not include the messages with severity `error`, `warning`, or `notice` that the T1600 routers log locally.

Table 138: Example: Local Severity Is notice, Forwarded Severity Is critical

Log Location	Source of Messages	Lowest Severity Included
T1600 router	Local	`notice`
TX Matrix Plus router	Local	`notice`
	Forwarded from T1600 routers	`critical`

Messages Logged When the Local Severity Level Is Higher

When the severity level is higher for local messages than for forwarded messages, the log on the TX Matrix Plus router includes fewer forwarded messages than when the severities are the same, and all local logs contain fewer messages overall.

For example, you can specify severity `critical` for the **/var/log/messages** file and severity `notice` for forwarded messages:

```
[edit system syslog]
file messages {
    any critical;
}
host sfc0-master {
    any notice;
}
```

Table 139 on page 1302 specifies which messages are included in the logs on the T1600 routers and the TX Matrix Plus router. Although the T1600 routers forward messages with severity `notice` and higher, the TX Matrix Plus router discards any of those messages with severity lower than `critical` (does not log forwarded messages with severity `error`, `warning`, or `notice`). None of the logs include messages with severity `error` or lower.

Table 139: Example: Local Severity Is critical, Forwarded Severity Is notice

Log Location	Source of Messages	Lowest Severity Included
T1600 router	Local	`critical`
TX Matrix Plus router	Local	`critical`
	Forwarded from T1600 routers	`critical`

Configuring Optional Features for Forwarded Messages on a TX Matrix Router

IN THIS SECTION

- Including Priority Information in Forwarded Messages | **1303**
- Adding a Text String to Forwarded Messages | **1303**
- Using Regular Expressions to Refine the Set of Forwarded Messages | **1304**

To configure additional optional features when specifying how the T640 routers forward messages to the TX Matrix router, include statements at the `[edit system syslog host scc-master]` hierarchy level. To include priority information (facility and severity level) in each forwarded message, include the `explicit-priority` statement. To insert a text string in each forwarded message, include the `log-prefix` statement. To use regular expression matching to specify more exactly which messages from a facility are forwarded, include the `match` statement.

```
[edit system syslog]
host scc-master {
    facility severity;
    explicit-priority;
    log-prefix string;
```

```
    match "regular-expression";
}
```

You can also include the `facility-override` statement at the `[edit system syslog host scc-master]` hierarchy level, but we do not recommend doing so. It is not necessary to use alternative facilities for messages forwarded to the TX Matrix router, because it runs the Junos system logging utility and can interpret the Junos OS-specific facilities. For more information about alternative facilities, see "Changing the Alternative Facility Name for System Log Messages Directed to a Remote Destination" on page 1318.

Including Priority Information in Forwarded Messages

When you include the `explicit-priority` statement at the `[edit system syslog host scc-master]` hierarchy level, messages forwarded to the TX Matrix router include priority information. For the information to appear in a log file on the TX Matrix router, you must also include the `explicit-priority` statement at the `[edit system syslog file filename]` hierarchy level for the file on the TX Matrix router. As a consequence, the log file with the same name on each platform in the routing matrix also includes priority information for locally generated messages.

To include priority information in messages directed to a remote machine from all routers in the routing matrix, also include the `explicit-priority` statement at the `[edit system syslog host hostname]` hierarchy level for the remote machine. For more information, see "Directing Messages to a Remote Destination from the Routing Matrix Based on the TX Matrix Router" on page 1323.

In the following example, the **/var/log/messages** file on all routers includes priority information for messages with severity `notice` and higher from all facilities. The log on the TX Matrix router also includes messages with those characteristics forwarded from the T640 routers.

```
[edit system syslog]
host scc-master {
    any notice;
    explicit-priority;
}
file messages {
    any notice;
    explicit-priority;
}
```

Adding a Text String to Forwarded Messages

When you include the `log-prefix` statement at the `[edit system syslog host scc-master]` hierarchy level, the string that you define appears in every message forwarded to the TX Matrix router. For more information, see "Adding a Text String to System Log Messages Directed to a Remote Destination" on page 1317.

Using Regular Expressions to Refine the Set of Forwarded Messages

When you include the `match` statement at the `[edit system syslog host scc-master]` hierarchy level, the regular expression that you specify controls which messages from the T640 routers are forwarded to the TX Matrix router. The regular expression is not applied to messages from the T640 router that are directed to destinations other than the TX Matrix router. For more information about regular expression matching, see "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

Configuring Optional Features for Forwarded Messages on a TX Matrix Plus Router

IN THIS SECTION

- Including Priority Information in Forwarded Messages | **1305**
- Adding a Text String to Forwarded Messages | **1305**
- Using Regular Expressions to Refine the Set of Forwarded Messages | **1306**

From the perspective of the user interface, the routing matrix appears as a single router. The TX Matrix Plus router (also called the switch-fabric chassis SFC) controls all the T1600 or T4000 routers (also called the line-card chassis LCC) in the routing matrix.

To configure additional optional features when specifying how the connected T1600 or T4000 LCCs forward messages to the SFC, include statements at the `[edit system syslog host sfc0-master]` hierarchy level. To include priority information (facility and severity level) in each forwarded message, include the `explicit-priority` statement. To insert a text string in each forwarded message, include the `log-prefix` statement. To use regular expression matching to specify more exactly which messages from a facility are forwarded, include the `match` statement.

```
[edit system syslog]
host sfc0-master {
    facility severity;
    explicit-priority;
    log-prefix string;
```

```
    match "regular-expression;
}
```

You can also include the facility-override statement at the [edit system syslog host sfc0-master] hierarchy level, but we do not recommend doing so. It is not necessary to use alternative facilities for messages forwarded to the SFC, because it runs the Junos system logging utility and can interpret the Junos OS-specific facilities. For more information about alternative facilities, see "Changing the Alternative Facility Name for System Log Messages Directed to a Remote Destination" on page 1318.

Including Priority Information in Forwarded Messages

When you include the explicit-priority statement at the [edit system syslog host sfc0-master] hierarchy level, messages forwarded to the TX Matrix Plus router (or the SFC) include priority information. For the information to appear in a log file on the SFC, you must also include the explicit-priority statement at the [edit system syslog file *filename*] hierarchy level for the file on the SFC. As a consequence, the log file with the same name on each platform in the routing matrix also includes priority information for locally generated messages.

To include priority information in messages directed to a remote machine from all routers in the routing matrix, also include the explicit-priority statement at the [edit system syslog host *hostname*] hierarchy level for the remote machine. For more information, see "Directing Messages to a Remote Destination from the Routing Matrix Based on a TX Matrix Plus Router" on page 1324.

In the following example, the **/var/log/messages** file on all routers includes priority information for messages with severity notice and higher from all facilities. The log on the TX Matrix Plus router SFC also includes messages with those characteristics forwarded from the connected T1600 or T4000 LCCs.

```
[edit system syslog]
host sfc0-master {
    any notice;
    explicit-priority;
}
file messages {
    any notice;
    explicit-priority;
}
```

Adding a Text String to Forwarded Messages

When you include the log-prefix statement at the [edit system syslog host sfc0-master] hierarchy level, the string that you define appears in every message forwarded to the TX Matrix Plus router. For more information, see "Adding a Text String to System Log Messages Directed to a Remote Destination" on page 1317.

Using Regular Expressions to Refine the Set of Forwarded Messages

When you include the `match` statement at the `[edit system syslog host sfc0-master]` hierarchy level, the regular expression that you specify controls which messages from the connected T1600 or T4000 LCCs are forwarded to the TX Matrix Plus SFC. The regular expression is not applied to messages from the connected LCCs that are directed to destinations other than the SFC. For more information about regular expression matching, see "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

Configuring System Logging Differently on Each T640 Router in a Routing Matrix

We recommend that all routers in a routing matrix composed of a TX Matrix router and T640 routers use the same configuration, which implies that you include system logging configuration statements on the TX Matrix router only. In rare circumstances, however, you might choose to log different messages on different routers. For example, if one router in the routing matrix is experiencing problems with authentication, a Juniper Networks support representative might instruct you to log messages from the `authorization` facility with severity `debug` on that router.

To configure routers separately, include configuration statements in the appropriate groups at the `[edit groups]` hierarchy level on the TX Matrix router:

- To configure settings that apply to the TX Matrix router but not the T640 routers, include them in the `re0` and `re1` configuration groups.
- To configure settings that apply to particular T640 routers, include them in the `lcc`*n*`-re0` and `lcc`*n*`-re1` configuration groups, where *n* is the line-card chassis (LCC) index number of the router.

When you use configuration groups, do not issue CLI configuration-mode commands to change statements at the `[edit system syslog]` hierarchy level on the TX Matrix router. If you do, the resulting statements overwrite the statements defined in configuration groups and apply to the T640 routers also. (We further recommend that you do not issue CLI configuration-mode commands on the T640 routers at any time.)

The following example shows how to configure the **/var/log/messages** files on three routers to include different sets of messages:

- On the TX Matrix router, local messages with severity `info` and higher from all facilities. The file does not include messages from the T640 routers, because the `host scc-master` statement disables message forwarding.
- On the T640 router designated `LCC0`, messages from the `authorization` facility with severity `info` and higher.

- On the T640 router designated `LCC1`, messages with severity `notice` from all facilities.

```
[edit groups]
re0 {
    system {
        syslog {
            file messages {
                any info;
            }
            host scc-master {
                any none;
            }
        }
    }
}
re1 {
    ... same statements as for re0 ...
}
lcc0-re0 {
    system {
        syslog {
            file messages {
                authorization info;
            }
        }
    }
}
lcc0-re1 {
    ... same statements as for lcc0-re0 ...
}
lcc1-re0 {
    system {
        syslog {
            file messages {
                any notice;
            }
        }
    }
}
lcc0-re1 {
```

```
    ... same statements as for lcc1-re0 ...
}
```

Configuring System Logging Differently on Each T1600 or T4000 Router in a Routing Matrix

We recommend that all routers in a routing matrix composed of a TX Matrix Plus router with T1600 or T4000 routers use the same configuration, which implies that you include system logging configuration statements on the TX Matrix Plus router only. In rare circumstances, however, you might choose to log different messages on different routers. For example, if one router in the routing matrix is experiencing problems with authentication, a Juniper Networks support representative might instruct you to log messages from the `authorization` facility with severity `debug` on that router.

To configure routers separately, include configuration statements in the appropriate groups at the `[edit groups]` hierarchy level on the TX Matrix Plus router:

- To configure settings that apply to the TX Matrix Plus router but not the T1600 or T4000 routers, include them in the `re0` and `re1` configuration groups.
- To configure settings that apply to particular T1600 or T4000 routers, include them in the `lcc`*n*`-re0` and `lcc`*n*`-re1` configuration groups, where *n* is the line-card chassis (LCC) index number of the router.

When you use configuration groups, do not issue CLI configuration-mode commands to change statements at the `[edit system syslog]` hierarchy level on the TX Matrix Plus router. If you do, the resulting statements overwrite the statements defined in configuration groups and apply to the T1600 or T4000 routers also. (We further recommend that you do not issue CLI configuration-mode commands on the T1600 or T4000 routers at any time.)

For more information about the configuration groups for a routing matrix, see the chapter about configuration groups in the *Junos OS CLI User Guide* .

The following example shows how to configure the **/var/log/messages** files on three routers to include different sets of messages:

- On the TX Matrix Plus router, local messages with severity `info` and higher from all facilities. The file does not include messages from the T1600 or T4000 routers, because the `host sfc0-master` statement disables message forwarding.
- On the T1600 or T4000 router designated `LCC0`, messages from the `authorization` facility with severity `info` and higher.

- On the T1600 or T4000 router designated `LCC1`, messages with severity `notice` from all facilities.

```
[edit groups]
re0 {
    system {
        syslog {
            file messages {
                any info;
            }
            host sfc0-master {
                any none;
            }
        }
    }
}
re1 {
    ... same statements as for re0 ...
}
lcc0-re0 {
    system {
        syslog {
            file messages {
                authorization info;
            }
        }
    }
}
lcc0-re1 {
    ... same statements as for lcc0-re0 ...
}
lcc1-re0 {
    system {
        syslog {
            file messages {
                any notice;
            }
        }
    }
}
lcc0-re1 {
```

```
    ... same statements as for lcc1-re0 ...
}
```

Direct System Log Messages to a Remote Destination

IN THIS SECTION

- Specify the Facility and Severity of Messages to Include in the Log | **1310**
- Direct System Log Messages to a Log File | **1313**
- Direct System Log Messages to a User Terminal | **1314**
- Direct System Log Messages to the Console | **1315**
- Direct System Log Messages to a Remote Machine or the Other Routing Engine | **1315**
- Specify an Alternative Source Address for System Log Messages Directed to a Remote Destination | **1316**
- Add a Text String to System Log Messages Directed to a Remote Destination | **1317**
- Change the Alternative Facility Name for System Log Messages Directed to a Remote Destination | **1318**
- Default Facilities for System Log Messages Directed to a Remote Destination | **1320**
- Alternate Facilities for System Log Messages Directed to a Remote Destination | **1320**
- Examples: Assign an Alternative Facility to System Log Messages Directed to a Remote Destination | **1322**
- Direct Messages to a Remote Destination from the Routing Matrix Based on the TX Matrix Router | **1323**
- Direct Messages to a Remote Destination from the Routing Matrix Based on a TX Matrix Plus Router | **1324**

Specify the Facility and Severity of Messages to Include in the Log

Each system log message belongs to a facility, which groups together messages that either are generated by the same source (such as a software process) or concern a similar condition or activity (such as authentication attempts). Each message is also preassigned a *severity level*, which indicates how seriously the triggering event affects routing platform functions.

When you configure logging for a facility and destination, you specify a severity level for each facility. Messages from the facility that are rated at that level and higher are logged to the following destination:

```
[edit system syslog]
(console | file filename | host destination | user username) {
    facility severity ;
}
```

For more information about the destinations, see "Directing System Log Messages to a User Terminal" on page 1314, and, "Directing System Log Messages to the Console" on page 1315.

To log messages belonging to more than one facility to a particular destination, specify each facility and associated severity as a separate statement within the set of statements for the destination.

Table 140 on page 1311 lists the Junos OS system logging facilities that you can specify in configuration statements at the `[edit system syslog]` hierarchy level.

Table 140: Junos OS System Logging Facilities

Facility	Type of Event or Error
any	All (messages from all facilities)
authorization	Authentication and authorization attempts
change-log	Changes to the Junos OS configuration
conflict-log	Specified configuration is invalid on the router type
daemon	Actions performed or errors encountered by system processes
dfc	Events related to dynamic flow capture
explicit-priority	Include priority and facility in system log messages
external	Actions performed or errors encountered by the local external applications

Table 140: Junos OS System Logging Facilities *(Continued)*

Facility	Type of Event or Error
`firewall`	Packet filtering actions performed by a firewall filter
`ftp`	Actions performed or errors encountered by the FTP process
`interactive-commands`	Commands issued at the Junos OS command-line interface (CLI) prompt or by a client application such as a Junos XML protocol or NETCONF XML client
`kernel`	Actions performed or errors encountered by the Junos OS kernel
`ntp`	Actions performed or errors encountered by the Network Time Protocol processes
`pfe`	Actions performed or errors encountered by the Packet Forwarding Engine
`security`	Security related events or errors
`user`	Actions performed or errors encountered by user-space processes

Table 141 on page 1312 lists the severity levels that you can specify in configuration statements at the `[edit system syslog]` hierarchy level. The levels from `emergency` through `info` are in order from highest severity (greatest effect on functioning) to lowest.

Unlike the other severity levels, the `none` level disables logging of a facility instead of indicating how seriously a triggering event affects routing functions. For more information, see "Disabling the System Logging of a Facility" on page 1285.

Table 141: System Log Message Severity Levels

Value	Severity Level	Description
N/A	`none`	Disables logging of the associated facility to a destination
0	`emergency`	System panic or other condition that causes the router to stop functioning

Table 141: System Log Message Severity Levels *(Continued)*

Value	Severity Level	Description
1	`alert`	Conditions that require immediate correction, such as a corrupted system database
2	`critical`	Critical conditions, such as hard errors
3	`error`	Error conditions that generally have less serious consequences than errors at the emergency, alert, and critical levels
4	`warning`	Conditions that warrant monitoring
5	`notice`	Conditions that are not errors but might warrant special handling
6	`info`	Events or nonerror conditions of interest
7	`any`	Includes all severity levels

Direct System Log Messages to a Log File

To direct system log messages to a file in the **/var/log** directory of the local Routing Engine, include the `file` statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
file filename {
    facility severity;
    archive <archive-sites (ftp-url  <password password>)> <files number> <size size> <start-
time "YYYY-MM-DD.hh:mm"> <transfer-interval minutes> <world-readable | no-world-readable>;
    explicit-priority;
    match "regular-expression";
    structured-data {
        brief;
```

```
    }
}
```

For the list of facilities and severity levels, see "Specifying the Facility and Severity of Messages to Include in the Log" on page 1310.

To prevent log files from growing too large, the Junos OS system logging utility by default writes messages to a sequence of files of a defined size. By including the `archive` statement, you can configure the number of files, their maximum size, and who can read them, either for all log files or for a certain log file. For more information, see "Specifying Log File Size, Number, and Archiving Properties" on page 1274.

For information about the following statements, see the indicated sections:

- `explicit-priority`—See "Including Priority Information in System Log Messages" on page 1276
- `match`—See "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281
- `structured-data`—See "Logging Messages in Structured-Data Format" on page 1274

Direct System Log Messages to a User Terminal

To direct system log messages to the terminal session of one or more specific users (or all users) when they are logged in to the local Routing Engine, include the `user` statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
user (username | *) {
    facility severity;
    match "regular-expression";
}
```

Specify one or more Junos OS usernames, separating multiple values with spaces, or use the asterisk (*) to indicate all users who are logged in to the local Routing Engine.

For the list of logging facilities and severity levels, see "Specifying the Facility and Severity of Messages to Include in the Log" on page 1310. For information about the `match` statement, see "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

Direct System Log Messages to the Console

To direct system log messages to the console of the local Routing Engine, include the `console` statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
console {
    facility severity;
}
```

For the list of logging facilities and severity levels, see "Specifying the Facility and Severity of Messages to Include in the Log" on page 1310.

Direct System Log Messages to a Remote Machine or the Other Routing Engine

To direct system log messages to a remote machine or to the other Routing Engine, include the `host` statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
host (hostname | other-routing-engine) {
    facility severity;
    explicit-priority;
    facility-override facility;
    log-prefix string;
    match "regular-expression";
    source-address source-address;
    structured-data {
        brief;
    }
}
source-address source-address;
```

To direct system log messages to a remote machine, include the `host` *hostname* statement to specify the remote machine's IP version 4 *(IPv4)* address, IP version 6 *(IPv6)* address, or fully qualified hostname. The remote machine must be running the standard `syslogd` utility. We do not recommend directing messages to another Juniper Networks device. In each system log message directed to the remote

machine, the hostname of the local *Routing Engine* appears after the timestamp to indicate that it is the source for the message.

To direct system log messages to the other Routing Engine on a device with two Routing Engines installed and operational, include the `host other-routing-engine` statement. The statement is not automatically reciprocal, so you must include it in each Routing Engine configuration if you want the Routing Engines to direct messages to each other. In each message directed to the other Routing Engine, the string `re0` or `re1` appears after the timestamp to indicate the source for the message.

For the list of logging facilities and severity levels to configure under the `host` statement, see "Specifying the Facility and Severity of Messages to Include in the Log" on page 1310.

To record facility and severity level information in each message, include the `explicit-priority` statement. For more information, see "Including Priority Information in System Log Messages" on page 1276.

For information about the `match` statement, see "Using Strings and Regular Expressions to Refine the Set of Logged Messages" on page 1281.

When directing messages to remote machines, you can include the `source-address` statement to specify the IP address of the device that is reported in the messages as their source. In each `host` statement, include the `facility-override` statement to assign an alternative facility and the `log-prefix` statement to add a string to each message. You can include the `structured-data` statement to enable the forwarding of structured system log messages to a remote system log server in the *IETF* system log message format.

Specify an Alternative Source Address for System Log Messages Directed to a Remote Destination

To specify the source router to be reported in *system log* messages when the messages are directed to a remote machine, include the `source-address` statement at the `[edit system syslog]` hierarchy level:

```
[edit system syslog]
source-address source-address;
```

source-address is a valid *IPv4* or *IPv6* address configured on one of the router interfaces. The address is reported in the messages directed to all remote machines specified in `host hostname` statements at the `[edit system syslog]` hierarchy level, but not in messages directed to the other Routing Engine.

Add a Text String to System Log Messages Directed to a Remote Destination

To add a text string to every system log message directed to a remote machine or to the other Routing Engine, include the `log-prefix` statement at the `[edit system syslog host]` hierarchy level:

```
[edit system syslog host (hostname | other-routing-engine)]
facility severity;
log-prefix string;
```

The string can contain any alphanumeric or special character except the equal sign (=) and the colon (:). It also cannot include the space character; do not enclose the string in quotation marks (“ ”) in an attempt to include spaces in it.

The Junos OS system logging utility automatically appends a colon and a space to the specified string when the system log messages are written to the log. The string is inserted after the identifier for the Routing Engine that generated the message.

The following example shows how to add the string M120 to all messages to indicate that the router is an M120 router, and direct the messages to the remote machine hardware-logger.mycompany.com:

```
[edit system syslog]
host hardware-logger.mycompany.com {
    any info;
    log-prefix M120;
}
```

When these configuration statements are included on an M120 router called origin1, a message in the system log on hardware-logger.mycompany.com looks like the following:

```
Mar 9 17:33:23 origin1 M120: mgd[477]: UI_CMDLINE_READ_LINE: user ‘root’, command ‘run show
version’
```

Change the Alternative Facility Name for System Log Messages Directed to a Remote Destination

Some facilities assigned to messages logged on the local router or switch have Junos OS-specific names (see "Junos OS System Logging Facilities" on page 1254). In the recommended configuration, a remote machine designated at the `[edit system syslog host hostname]` hierarchy level is not a Juniper Networks router or switch, so its syslogd utility cannot interpret the Junos OS-specific names. To enable the standard syslogd utility to handle messages from these facilities when messages are directed to a remote machine, a standard `localX` facility name is used instead of the Junos OS-specific facility name.

"Default Facilities for System Log Messages Directed to a Remote Destination" on page 1320 lists the default alternative facility name next to the Junos OS-specific facility name it is used for.

The syslogd utility on a remote machine handles all messages that belong to a facility in the same way, regardless of the source of the message (the Juniper Networks router or switch or the remote machine itself). For example, the following statements in the configuration of the router called `local-router` direct messages from the `authorization` facility to the remote machine *monitor.mycompany.com*:

```
[edit system syslog]
host monitor.mycompany.com {
    authorization info;
}
```

The default alternative facility for the local `authorization` facility is also `authorization`. If the syslogd utility on `monitor` is configured to write messages belonging to the `authorization` facility to the file **/var/log/auth-attempts**, then the file contains the messages generated when users log in to `local-router` and the messages generated when users log in to `monitor`. Although the name of the source machine appears in each system log message, the mixing of messages from multiple machines can make it more difficult to analyze the contents of the `auth-attempts` file.

To make it easier to separate the messages from each source, you can assign an alternative facility to all messages generated on `local-router` when they are directed to `monitor`. You can then configure the syslogd utility on `monitor` to write messages with the alternative facility to a different file from messages generated on `monitor` itself.

To change the facility used for all messages directed to a remote machine, include the `facility-override` statement at the `[edit system syslog host hostname]` hierarchy level:

```
[edit system syslog host hostname]
facility severity;
facility-override facility;
```

In general, it makes sense to specify an alternative facility that is not already in use on the remote machine, such as one of the `local`*X* facilities. On the remote machine, you must also configure the syslogd utility to handle the messages in the desired manner.

"Facilities for the facility-override Statement" on page 1310 lists the facilities that you can specify in the `facility-override` statement.

We do not recommend including the `facility-override` statement at the `[edit system syslog host other-routing-engine]` hierarchy level. It is not necessary to use alternative facility names when directing messages to the other Routing Engine, because its Junos OS system logging utility can interpret the Junos OS-specific names.

The following example shows how to log all messages generated on the local router at the error level or higher to the local0 facility on the remote machine called monitor.mycompany.com:

```
[edit system syslog]
host monitor.mycompany.com {
    any error;
    facility-override local0;
}
```

The following example shows how to configure routers located in California and routers located in New York to send messages to a single remote machine called central-logger.mycompany.com. The messages from California are assigned to alternative facility local0 and the messages from New York are assigned to alternative facility local2.

- Configure California routers to aggregate messages in the local0 facility:

  ```
  [edit system syslog]
  host central-logger.mycompany.com {
      change-log info;
      facility-override local0;
  }
  ```

- Configure New York routers to aggregate messages in the local2 facility:

  ```
  [edit system syslog]
  host central-logger.mycompany.com {
      change-log info;
      facility-override local2;
  }
  ```

On central-logger, you can then configure the system logging utility to write messages from the local0 facility to the file **change-log** and the messages from the local2 facility to the file **new-york-config**.

Default Facilities for System Log Messages Directed to a Remote Destination

Table 142 on page 1320 lists the default alternative facility name next to the Junos OS-specific facility name for which it is used. For facilities that are not listed, the default alternative name is the same as the local facility name.

Table 142: Default Facilities for Messages Directed to a Remote Destination

Junos OS–Specific Local Facility	Default Facility When Directed to Remote Destination
change-log	local6
conflict-log	local5
dfc	local1
firewall	local3
interactive-commands	local7
pfe	local4

Alternate Facilities for System Log Messages Directed to a Remote Destination

Table 143 on page 1321 lists the facilities that you can specify in the `facility-override` statement.

Table 143: Facilities for the facility-override Statement

Facility	Description
`authorization`	Authentication and authorization attempts
`daemon`	Actions performed or errors encountered by system processes
`ftp`	Actions performed or errors encountered by the FTP process
`kernel`	Actions performed or errors encountered by the Junos OS kernel
`local0`	Local facility number 0
`local1`	Local facility number 1
`local2`	Local facility number 2
`local3`	Local facility number 3
`local4`	Local facility number 4
`local5`	Local facility number 5
`local6`	Local facility number 6
`local7`	Local facility number 7
`user`	Actions performed or errors encountered by user-space processes

We do not recommend including the `facility-override` statement at the `[edit system syslog host other-routing-engine]` hierarchy level. It is not necessary to use alternative facility names when directing messages to the other Routing Engine, because its Junos OS system logging utility can interpret the Junos OS-specific names.

Examples: Assign an Alternative Facility to System Log Messages Directed to a Remote Destination

Log all messages generated on the local routing platform at the error level or higher to the `local0` facility on the remote machine called `monitor.mycompany.com`:

```
[edit system syslog]
host monitor.mycompany.com {
    any error;
    facility-override local0;
}
```

Configure routing platforms located in California and routing platforms located in New York to send messages to a single remote machine called central-logger.mycompany.com. The messages from California are assigned alternative facility local0 and the messages from New York are assigned to alternative facility local2.

- Configure California routing platforms to aggregate messages in the `local0` facility:

  ```
  [edit system syslog]
  host central-logger.mycompany.com {
      change-log info;
      facility-override local0;
  }
  ```

- Configure New York routing platforms to aggregate messages in the `local2` facility:

  ```
  [edit system syslog]
  host central-logger.mycompany.com {
      change-log info;
      facility-override local2;
  }
  ```

On `central-logger`, you can then configure the system logging utility to write messages from the `local0` facility to the file **california-config** and the messages from the `local2` facility to the file **new-york-config**.

Direct Messages to a Remote Destination from the Routing Matrix Based on the TX Matrix Router

You can configure a routing matrix composed of a TX Matrix router and T640 routers to direct system logging messages to a remote machine or the other Routing Engine on each router, just as on a single-chassis system. Include the `host` statement at the `[edit system syslog]` hierarchy level on the TX Matrix router:

```
[edit system syslog]
host (hostname | other-routing-engine) {
    facility severity;
    explicit-priority;
    facility-override facility;
    log-prefix string;
    match "regular-expression";
}
source-address source-address;
```

The TX Matrix router directs messages to a remote machine or the other Routing Engine in the same way as a single-chassis system, and the optional statements (`explicit-priority`, `facility-override`, `log-prefix`, `match`, and `source-address`) also have the same effect as on a single-chassis system.

For the TX Matrix router to include priority information when it directs messages that originated on a T640 router to the remote destination, you must also include the `explicit-priority` statement at the `[edit system syslog host scc-master]` hierarchy level.

The `other-routing-engine` statement does not interact with message forwarding from the T640 routers to the TX Matrix router. For example, if you include the statement in the configuration for the Routing Engine in slot 0 (`re0`), the `re0` Routing Engine on each T640 router sends messages to the `re1` Routing Engine on its platform only. It does not also send messages directly to the `re1` Routing Engine on the TX Matrix router.

Because the configuration on the TX Matrix router applies to the T640 routers, any T640 router that has interfaces for direct access to the Internet also directs messages to the remote machine. The consequences include the following:

- If the T640 routers are configured to forward messages to the TX Matrix router (as in the default configuration), the remote machine receives two copies of some messages: one directly from the T640 router and the other from the TX Matrix router. Which messages are duplicated depends on whether the severities are the same for local logging and for forwarded messages. For more information, see "Configuring Message Forwarding to the TX Matrix Router" on page 1294.

- If the source-address statement is configured at the [edit system syslog] hierarchy level, all routers in the routing matrix report the same source address in messages directed to the remote machine. This is appropriate, because the routing matrix functions as a single router.
- If the log-prefix statement is included, the messages from all routers in the routing matrix include the same text string. You cannot use the string to distinguish between the routers in the routing matrix.

Direct Messages to a Remote Destination from the Routing Matrix Based on a TX Matrix Plus Router

From the perspective of the user interface, the routing matrix appears as a single router. The TX Matrix Plus router (also called the switch-fabric chassis SFC) controls all the T1600 or T4000 routers also called the ine-card chassis LCC) in the routing matrix.

You can configure a routing matrix composed of a TX Matrix Plus router with connected T1600 or T4000 LCCs to direct system logging messages to a remote machine or the other Routing Engine on each routing router, just as on a single-chassis system. Include the host statement at the [edit system syslog] hierarchy level on the SFC:

```
[edit system syslog]
host (hostname | other-routing-engine) {
    facility severity;
    explicit-priority;
    facility-override facility;
    log-prefix string;
    match "regular-expression";
}
source-address source-address;
```

The TX Matrix Plus router directs messages to a remote machine or the other Routing Engine in the same way as a single-chassis system, and the optional statements (explicit-priority, facility-override, log-prefix, match, and source-address) also have the same effect as on a single-chassis system.

For the TX Matrix Plus router to include priority information when it directs messages that originated on a connected T1600 or T4000 LCC to the remote destination, you must also include the explicit-priority statement at the [edit system syslog host sfc0-master] hierarchy level.

The other-routing-engine statement does not interact with message forwarding from the connected T1600 or T4000 LCCs to the SFC. For example, if you include the statement in the configuration for the Routing Engine in slot 0 (re0), the re0 Routing Engine on each connected T1600 or T4000 LCC sends

messages to the re1 Routing Engine on its router only. It does not also send messages directly to the re1 Routing Engine on the SFC.

Because the configuration on the SFC applies to the connected T1600 or T4000 LCCs, any LCC that has interfaces for direct access to the Internet also directs messages to the remote machine. The consequences include the following:

- If the LCCs are configured to forward messages to the SFC (as in the default configuration), the remote machine receives two copies of some messages: one directly from the T1600 or T4000 LCC and the other from the SFC. Which messages are duplicated depends on whether the severities are the same for local logging and for forwarded messages. For more information, see "Configuring Message Forwarding to the TX Matrix Plus Router" on page 1295.
- If the source-address statement is configured at the [edit system syslog] hierarchy level, all routers in the routing matrix report the same source address in messages directed to the remote machine. This is appropriate, because the routing matrix functions as a single routing router.
- If the log-prefix statement is included, the messages from all routers in the routing matrix include the same text string. You cannot use the string to distinguish between the routers in the routing matrix.

Display System Log Files

IN THIS SECTION

- Display a Log File from a Single-Chassis System | **1326**
- Log File Sample Content | **1326**
- Display a Log File from a Routing Matrix | **1328**
- Display MD5 Log Files | **1329**

Display a Log File from a Single-Chassis System

To display a log file stored on a single-chassis system, enter Junos OS CLI operational mode and issue either of the following commands:

```
user@host> show log log-filename
user@host> file show log-file-pathname
```

By default, the commands display the file stored on the local Routing Engine. To display the file stored on a particular Routing Engine, prefix the file or pathname with the string `re0` or `re1` and a colon. The following examples both display the **/var/log/messages** file stored on the Routing Engine in slot 1:

```
user@host> show log re1:messages
 user@host> file show re1:/var/log/messages
```

For information about the fields in a log message, see Interpreting Messages Generated in Standard Format by a Junos OS Process or Library, Interpreting Messages Generated in Standard Format by Services on a PIC, and Interpreting Messages Generated in Structured-Data Format. For examples, see "Log File Sample Content" on page 1326.

Log File Sample Content

This topic contains sample content from the /var/log directory. You can display the contents of the **/var/log/messages** file stored on the local Routing Engine. (The **/var/log** directory is the default location for log files, so you do not need to include it in the filename. The **messages** file is a commonly configured destination for system log messages.)

> **NOTE**: In Junos OS Evolved, the **messages** file is only written on the primary Routing Engine. Backup Routing Engine messages are found in the **messages** file on the primary Routing Engine.

```
user@host> show log messages Apr 11 10:27:25 router1 mgd[3606]: UI_DBASE_LOGIN_EVENT: User
'barbara' entering configuration mode
 Apr 11 10:32:22 router1 mgd[3606]: UI_DBASE_LOGOUT_EVENT: User 'barbara' exiting configuration
mode
Apr 11 11:36:15 router1 mgd[3606]: UI_COMMIT: User 'root' performed commit: no comment
```

```
 Apr 11 11:46:37 router1 mib2d[2905]: SNMP_TRAP_LINK_DOWN: ifIndex 82, ifAdminStatus up(1),
ifOperStatus down(2), ifName at-1/0/0
```

You can display the contents of the file **/var/log/processes**, which has been previously configured to include messages from the `daemon` facility. When issuing the `file show` command, you must specify the full pathname of the file:

```
user@host> file show /var/log/processes Feb 22 08:58:24 router1 snmpd[359]:
SNMPD_TRAP_WARM_START: trap_generate_warm: SNMP trap: warm start
 Feb 22 20:35:07 router1 snmpd[359]: SNMPD_THROTTLE_QUEUE_DRAINED: trap_throttle_timer_handler:
cleared all throttled traps
 Feb 23 07:34:56 router1 snmpd[359]: SNMPD_TRAP_WARM_START: trap_generate_warm: SNMP trap: warm
start
 Feb 23 07:38:19 router1 snmpd[359]: SNMPD_TRAP_COLD_START: trap_generate_cold: SNMP trap: cold
start
```

You can display the contents of the file **/var/log/processes** when the `explicit-priority` statement is included at the [`edit system syslog file processes`] hierarchy level:

```
user@host> file show /var/log/processes Feb 22 08:58:24 router1 snmpd[359]:
%DAEMON-3-SNMPD_TRAP_WARM_START: trap_generate_warm: SNMP trap: warm start
 Feb 22 20:35:07 router1 snmpd[359]:
%DAEMON-6-SNMPD_THROTTLE_QUEUE_DRAINED: trap_throttle_timer_handler: cleared all throttled traps
Feb 23 07:34:56 router1 snmpd[359]:
%DAEMON-3-SNMPD_TRAP_WARM_START: trap_generate_warm: SNMP trap: warm start
Feb 23 07:38:19 router1 snmpd[359]:
%DAEMON-2-SNMPD_TRAP_COLD_START: trap_generate_cold: SNMP trap: cold start
```

Warning Message Support for Throughput Overuse:

The SRX4100 device supports up to 20 Gbps and 7 Mpps of Internet mix (IMIX) firewall performance. When IMIX throughput exceeds 20 Gbps and 7 Mpps on an SRX4100 device, new log messages are logged. These log messages remind you that there is throughput overuse. You can see the following sample log messages when you issue the `show log messages` command.

```
user@host> show log messages
Apr 25 14:01:12 user Throughput exceed 20Gbps and 7Mpps in 35% of last 15 minutes, above the
time threshold 10%!
Apr 25 14:16:12 user Throughput exceed 20Gbps and 7Mpps in 95% of last 15 minutes, above the
time threshold 10%!
```

As a reminder of throughput overuse, every 15 minutes the system calculates how many minutes the throughout has exceeded 20 Gbps and 7 Mpps. The system triggers a log message if the throughput has exceeded more than 1 minute, 30 seconds (10%) of the last 15 minutes. For example, suppose you see the following log message:

```
Throughput exceed 20 Gbps and 7 Mpps in 35% of last 15 minutes, above the time threshold 10%!
```

It means your throughput has exceeded 20 Gbps and 7 Mpps for 5 minutes, 15 seconds of the last 15 minutes (35% of 15 minutes) that triggered the log message.

To turn off this log message, we recommend that you bring down the throughput level below 20 Gbps and 7 Mpps or install the enhanced performance upgrade license.

> **NOTE**: This feature requires a license. Please refer to the Juniper Licensing Guide for general information about License Management. Please refer to the product Data Sheets at SRX Series Services Gateways for details, or contact your Juniper Account Team or Juniper Partner.

Display a Log File from a Routing Matrix

One way to display a log file stored on the local Routing Engine of any of the individual platforms in a routing matrix (T640 routing nodes or TX Matrix platform) is to log in to a Routing Engine on the platform, enter Junos OS CLI operational mode, and issue the `show log` or `file show` command described in "Displaying a Log File from a Single-Chassis System" on page 1326.

To display a log file stored on a T640 routing node during a terminal session on the TX Matrix platform, issue the `show log` or `file show` command and add a prefix that specifies the T640 routing node's LCC index number as lccn, followed by a colon. The index can be from 0 (zero) through 3:

```
user@host> show log lccn:log-filename
user@host> file show lccn:log-file-pathname
```

By default, the `show log` and `file show` commands display the specified log file stored on the primary Routing Engine on the T640 routing node. To display the log from a particular Routing Engine, prefix the file- or pathname with the string lccn-primary, `lccn-re0`, or `lccn-re1`, followed by a colon. The following

examples all display the **/var/log/messages** file stored on the primary Routing Engine (in slot 0) on routing node LCC2:

```
user@host> show log lcc2:messages
 user@host> show log lcc2-master:messages
 user@host> show log lcc2-re0:messages
user@host> file show lcc2:/var/log/messages
```

If the T640 routing nodes are forwarding messages to the TX Matrix platform (as in the default configuration), another way to view messages generated on a T640 routing node during a terminal session on the TX Matrix platform is simply to display a local log file. However, the messages are intermixed with messages from other T640 routing nodes and the TX Matrix platform itself. For more information about message forwarding, see "Impact of Different Local and Forwarded Severity Levels on System Log Messages on a TX Matrix Router" on page 1296.

For information about the fields in a log message, see Interpreting Messages Generated in Structured-Data Format, Interpreting Messages Generated in Standard Format by Services on a PIC, and Interpreting Messages Generated in Standard Format by a Junos OS Process or Library. For examples, see "Log File Sample Content" on page 1326.

Display MD5 Log Files

Junos OS and Junos OS Evolved BGP supports authentication for protocol exchanges. When you configure TCP Message Digest 5 (MD5) authentication for BGP protocol on the neighboring routing devices to verify the authenticity of BGP packets, the following log warning messages stored in **/var/log/messages/** are displayed:

On Junos OS,

When MD5 configured on local but not on peer device,

```
Apr 16 21:49:52  R1_re kernel: tcp_auth_ok: Packet from 2.2.2.2:52848 missing MD5 digest
```

When MD5 configured on peer but not on local device,

```
Apr 16 21:51:30  R1_re kernel: tcp_auth_ok: Packet from 2.2.2.2:54049 unexpectedly has MD5 digest
```

When MD5 is configured on both the routers and there is authentication password mismatch, the following log is displayed:

```
Apr 16 21:51:58  R1_re kernel: tcp_auth_ok: Packet from 2.2.2.2:54049 wrong MD5 digest
```

On Junos OS Evolved,

When TCP MD5 authentication is configured on local but not on peer device, the log messages are not available.

When TCP MD5 authentication is configured on peer but not on local device, the log messages are not available.

When MD5 is configured on both the routers and there is authentication password mismatch, the following log is displayed:

```
Apr 16 21:41:22  vScapa1-RE0-re0 kernel: %KERN-6-TCP: MD5 Hash failed for (2.2.2.2, 39213)-
>(1.1.1.1, 179)
```

Configure System Logging for Security Devices

IN THIS SECTION

- System Logging Overview for Security Devices | **1331**
- Binary Format for Security Logs | **1333**
- On-Box Logging and Reporting | **1334**
- Monitor Reports | **1341**
- Configure On-Box Binary Security Log Files | **1352**
- Configure Off-Box Binary Security Log Files | **1354**
- Configure On-Box Protobuf Security Log Files in Event Mode | **1356**
- Configure On-Box Protobuf Security Log Files in Stream Mode | **1357**
- Configure Off-box Protobuf Security Log Files | **1359**
- Send System Log Messages to a File | **1360**
- Configure the System to Send All Log Messages Through eventd | **1361**

System Logging Overview for Security Devices

IN THIS SECTION

- Control Plane and Data Plane Logs | 1331
- Redundant System Log Server | 1332

Junos OS supports configuring and monitoring of system log messages (also called *syslog messages*). You can configure files to log system messages and also assign attributes, such as severity levels, to messages. Reboot requests are recorded to the system log files, which you can view with the `show log` command.

This section contains the following topics:

Control Plane and Data Plane Logs

Junos OS generates separate log messages to record events that occur on the system's control and data planes.

- The control plane logs, also called system logs, include events that occur on the routing platform. The system sends control plane events to the `eventd` process on the Routing Engine, which then handles the events by using Junos OS policies, by generating system log messages, or both. You can choose to send control plane logs to a file, user terminal, routing platform console, or remote machine. To generate control plane logs, use the `syslog` statement at the `[system]` hierarchy level.
- The data plane logs, also called *security logs*, primarily include security events that are handled inside the data plane. Security logs can be in text or binary format, and they can be saved locally (event mode) or sent to an external server (stream mode). Binary format is required for stream mode and recommended to conserve log space in event mode.

  Note the following:

  - Security logs can be saved locally (on box) or externally (off box), but not both.
  - SRX1400, SRX1500, SRX3400, SRX3600, SRX4100, SRX4200, SRX4600, SRX5400, SRX5600, and SRX5800 devices default to stream mode. To specify binary format and an external server, see "Configuring Off-Box Binary Security Log Files" on page 1354.

> **NOTE**: Logs might be dropped if you configure event mode logging on these devices.

Starting with Junos OS Release 15.1X49-D100, the default mode for SRX1500 device is stream mode. Prior to Junos OS Release 15.1X49-D100, the default mode for SRX1500 device was event mode.

- Apart from Junos OS Release 18.4R1, 18,4R2, 19.1 and 19.2R1, in all the other releases starting in Junos OS Release 18.3R3, the default logging mode for SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550M devices is stream mode. Data plane events are written to system log files in a similar manner to control plane events. To specify binary format for the security logs, see "Configuring Off-Box Binary Security Log Files" on page 1354.

Starting in Junos OS Release 20.2R1, we support escape in stream log forwarding and on-box reporting to avoid parsing errors. Stream mode supports escape in `sd-syslog` and `binary` formats when logs are not sent to `eventd` process. For the logs send to `eventd` process, we recommend not to enable an `escape` option as the `eventd` process has enabled the escape for the structure log. Event mode supports escape only in `binary` format. By default the `escape` option is disabled. You must enable the `escape` option using the `set security log escape` command.

Redundant System Log Server

Security system logging traffic intended for remote servers is sent through the network interface ports, which support two simultaneous system log destinations. Each system logging destination must be configured separately. When two system log destination addresses are configured, identical logs are sent to both destinations. While two destinations can be configured on any device that supports the feature, adding a second destination is primarily useful as a redundant backup for standalone and active/backup configured *chassis cluster* deployments.

The following redundant server information is available:

- Facility: `cron`
- Description: cron scheduling process
- Severity Level (from highest to lowest severity): `debug`
- Description: Software debugging messages

Binary Format for Security Logs

The Junos OS generates separate log messages to record events that occur on the system's control plane and data plane. The control plane monitors events that occur on the routing platform. Such events are recorded in system log messages. To generate system log messages, use the `syslog` statement at the `[system]` hierarchy level.

Data plane log messages, referred to as security log messages, record security events that the system handles directly inside the data plane. To generate security log messages, use the `log` statement at the `[security]` hierarchy level.

System log messages are maintained in log files in text-based formats, such as BSD Syslog, Structured Syslog, and WebTrends Enhanced Log Format (WELF).

Security log messages can also be maintained in text-based formats. Because security logging can produce large amounts of data, however, text-based log files can quickly consume storage and CPU resources. Depending on your implementation of security logging, a log file in a binary-based format can provide more efficient use of on-box or off-box storage and improved CPU utilization. Binary format for security log messages is available on all SRX Series Firewalls.

When configured in event mode, security log messages generated in the data plane are directed to the control plane and stored locally on the device. Security log messages stored in binary format are maintained in a log file separate from that used to maintain system log messages. Events stored in a binary log file are not accessible with advanced log-scripting commands intended for text-based log files. A separate CLI operational command supports decoding, converting, and viewing binary log files that are stored locally on the device.

When configured in stream mode, security log messages generated in the data plane are streamed to a remote device. When these messages are stored in binary format, they are streamed directly to an external log collection server in a Juniper-specific binary format. Externally-stored binary log files can only be read using Juniper Secure Analytics (JSA) or Security Threat Response Manager (STRM).

Starting in Junos OS Release 17.4R2 and later, on SRX300, SRX320, SRX340, SRX345 Series devices and vSRX Virtual Firewall instances, when the device is configured in stream mode, you can configure maximum of eight system log hosts. In Junos OS Release 17.4R2 and earlier releases, you can configure only three system log hosts in the stream mode. If you configure more than three system log hosts, then the following error message is displayed **error: configuration check-out failed**.

For information about configuring on-box (event-mode) binary security logs, please see "Configuring On-Box Binary Security Log Files" on page 1352. For information about configuring off-box (stream-mode) binary security logs, please see "Configuring Off-Box Binary Security Log Files" on page 1354.

On-Box Logging and Reporting

IN THIS SECTION

- Overview | 1334
- On-Box Reporting Features | 1338
- Table Selection | 1340
- Table Lifetime | 1340
- Table Dense Mode | 1341
- Chassis Cluster Scenario | 1341

This topic describes the on-box logging and reporting CLI functionality and the design aspects of on-box reporting for the SRX devices.

Overview

On-box traffic logging to solid-state drives (SSDs) supports eight external log servers or files.

An all-in-one XML file is added that contains all the traffic logs information. The XML file also generates all the logging header files and traffic log related documents.

A new process (daemon) called *local log management daemon (llmd)* is supported in Services Processing Cards 0 (SPCs0) to handle on-box traffic logging. Traffic produced by flowd in SPCs is listed in traffics logs. The llmd saves these logs to the local SSD. Traffic logs are saved in the four different formats. See Table 144 on page 1334 to know about the log formats.

Table 144: Log formats

Log format	Description	Default
Syslog	• Traditional log format to save logs.	Yes

Table 144: Log formats *(Continued)*

Log format	Description	Default
Sd-syslog	• Structured system log file format. • Most descriptive and lengthy hence takes more space to store. • Takes more time to transfer logs saved in this format because of the size.	-
Welf	• WebTrends Enhanced Log file Format is an industry standard log file exchange format. • Compatible with Firewall Suite 2.0 and later, Firewall Reporting Center 1.0 and later, and Security Reporting Center 2.0 and later.	-
Binary	• Juniper proprietary format. • Least descriptive among all the other log formats and takes the least space compared to other log formats.	-
protobuf	• Google's language-neutral, platform-neutral, extensible mechanism for serializing structured data. • A different method is used to encode the data. • File size is small compared to syslog and sd-syslog.	

On-box reporting mechanism is an enhancement to the existing logging functionality. The existing logging functionality is modified to collect system traffic logs, analyzes the logs, and generate reports of these logs in the form of tables using the CLI. On-box reporting feature is intended to provide a simple and easy to use interface for viewing security logs. The on-box reports are easy to use J-Web pages of various security events in the form of tables and graphs. The reports allow the IT security management to identify security information at a glance, and quickly decide the actions to be taken. Thorough analysis of logs is performed (based on session types) for features such as screen, IDP, Content Security and IPSec.

You can define filters for the log data that is reported on based on the following criteria:

NOTE: The top, in-detail, and in-interval conditions cannot be used at the same time.

- `top <number>`—This option allow you to generate reports for top security events as specified in the command. for example: top 5 IPS attacks or top 6 URLs detected through Content Security.
- `in-detail <number>`—This option allow you to generate detail log content.
- `in-interval <time-period>`—This option allows you to generate the events logged between certain time intervals.
- `summary`—This option allows you to generate the summary of the events. In this way, you can fine-tune the report to your needs, and displays only the data that you want to use.

The maximum in-interval number which shows the count in intervals is 30. If large duration is specified, then the counters are assembled to ensure the maximum in-interval is less than 30.

Both in-detail and summary have the “all” option, since different table have different attribute (like session table does not have the attribute “reason” but Content Security has), the “all” option does not have any filter except start-time and stop-time. If there is any other filter other than start time and stop time then an error is displayed.

For example: root@host> show security log report in-detail all reason reason1

```
error: "query condition error"
```

The application firewall logs for application and user visibility will list applications and nested applications. When the logs of these features list nested applications then nested applications are listed in J-Web. When the logs list nested applications as not-applicable or unknown then only the applications are listed in J-Web.

Use the following CLI commands for application and user visibility for all the applications and nested applications listing:

- For top nested-application by count—`show security log report top session-close top-number <number> group-by application order-by count with user`
- For top nested-application by volume—`show security log report top session-close top-number <number> group-by application order-by volume with user`
- For top user by count with nested application—`show security log report top session-close top-number <number> group-by user order-by count with application`

The on-box reporting feature is enabled by default when you load the factory-default configurations on the SRX Series Firewall with Junos OS Release 15.1X49-D100 or later.

If you are upgrading your SRX Series Firewall from a Junos OS Release prior to Junos OS 15.1X49-D100, then the SRX Series Firewall inherits the existing configuration and the on-box reporting feature is disabled by default. You need to configure the `set security log report` command and the `set security log mode stream` command to enable the on-box reporting feature on the device that are upgraded.

Starting in Junos OS Release 19.3R1 , the factory-default configuration does not include on-box reporting configuration to increase the solid-state drive (SSD) lifetime. You can enable the on-box reporting feature by configuring the `set security log report` CLI command at `[edit security log]` hierarchy.

See J-Web User Guide for SRX Series Devices to perform this task on J-Web user interface.

Starting in Junos OS Release 21.3R1, the on-box reporting logs are stored on the memory file system (MFS) if there is no external SSD. The maximum number of logs that you can save on MFS is lesser than what you can save on an external SSD. This prevents memory exhaustion and failure. Logs saved in MFS are not retained after device reboot or power failure. See Table 145 on page 1337 to know the number of logs recorded in on-box reporting and off-box reporting.

Table 145: Number of Logs

Reporting Mode	Session	Screen	IDP	Content Security	IPsec-VPN	SKY
Off-box	1200,000	120,000	120,000	120,000	40,000	120,000
On-box	500,000	50,000	50,000	50,000	20,000	50,000

NOTE: You must configure security policy for the session using the `set security policies from-zone zone-name to-zone zone-name policy` *`policy-name`* `then log session-close` command to list all the applications and nested applications in Application Tracking on J-Web using the on-box reporting feature. See for more *log (Security Policies)* for details.

After the log message is recorded, the log is stored within a log file which is then stored in the database table of the RE for further analysis (on SRX300, SRX320, SRX340, SRX345, and SRX550M devices) or on the SSD card for further analysis (on SRX1500, SRX4100, and SRX4200 devices).

NOTE: This feature supports receiving top most reports based on count or volume of the session or the type of log, captures events occurring in each second within a specified time range,

captures log content for a specified CLI condition. Various CLI conditions like “summary”, top”, “in-detail”, and “in-interval” are used to generate reports. You can generate only one report at one time using the CLI. All the CLI conditions cannot be used at the same time. You can generate only one report at one time using the CLI.

The benefits of this feature are:

- Reports are stored locally on the SRX Series Firewall and there is no requirement for separate devices or tools for logs and reports storage.
- The on-box reports are easy-to-use J-Web pages of various security events in the form of tables and graphs.
- Provides a simple and easy-to-use interface for viewing security logs.
- The reports generated enables the IT security management team to identify security information at a glance and quickly decide the actions to be taken.

The on-box reporting feature supports:

- Generating reports based on the requirements. For example: count or volume of the session, types of logs for activities such as IDP, Content Security, IPsec VPN.
- Capturing real-time events within a specified time range.
- Capturing all the network activities in a logical, organized, and easy-to-understand format based on various CLI specified conditions.

On-Box Reporting Features

The on-box reporting feature supports:

- **Sqlite3 support as a library**—sqlite3 was not supported prior to Junos OS release 15.1X49-D100. Starting with Junos OS Release 15.1X49-D100, an SQL log database (SQLite Version 3) is used by the daemons running on the RE as well as other potential modules to store logs on SRX Series Firewalls.

  In Junos OS Release 19.4R1, we've upgraded the on-box logging database to improve query performance.

- **Running llmd in both Junos OS and Linux OS**—The forwarding daemon (flowd) decodes database index from binary logs and sends both index and log to the local log management daemon (llmd).

On SRX300, SRX320, SRX340, SRX345, and SRX550M devices, llmd runs in Junos OS. On SRX1500, SRX4100, and SRX4200 devices, llmd runs in Linux. So, for supporting llmd to run in both Junos OS and Linux OS, the llmd code directory is moved from the Linux side to the Junos OS side.

- **Storing of logs into specified table of the sqlite3 database by llmd**— A new syslog daemon is introduced to collect local logs on SRX Series Firewalls and saving them into the database.

  Starting in Junos OS Release 19.3R1, Junos OS stores logs in multiple tables instead of a single table in a database file. Each table contains the timestamp of the oldest and latest logs. When you initiate a query based on the start and end time, llmd finds latest table to generate reports.

  For example, if there are 5 million logs in one table of a database file generated in the last 10 hours, and if you want to take a report, you should spend more than half an hour. From Junos OS Release 19.3R1, one table is separated into multiple tables, and each table has 0.5 million of logs. To generate the same report, one table information is sufficient.

  We recommend you to query with a shorter time for better performance.

  - **Database table definition**—For session logs, the data types are source-address, destination-address, application, user, and so on. For logs related to security features, the data types are attack-name, URL, profile protocol, and so on. Therefore, different tables are designed to store different types of logs to help improve the performance and save disk space. SRX Series Firewall creates a database table for each log type, when log data is recorded.

    Each type of database table has its maximum record number that is device specific. When the table record number reaches the limitation, new logs replace the oldest logs. Junos OS stores log in an SRX Series Firewall in which active traffic is passed.

    Starting in Junos OS Release 19.3R1, you can create multiple tables in a database file to store logs. You can define the capacity to store logs in a table.

    If the limit of log number exceeds the table capacity, Junos OS stores the logs in the second table. For example, if the limit of logs in table 1 exceeds the table capacity, Junos OS stores logs in table 2.

    If the limit of the log number exceeds the last table of file 1, Junos OS stores the logs in table 1 of file 2. For example, table n is the last table of file 1. When the logs exceed the table capacity, Junos OS stores the logs in table 1 of file 2.

    To make immediate effect after you change the table number, use `clear security log report` operational command.

  - **Database table rotation**—Each type of database table has its maximum record number that is device specific. When the table record number reaches the limitation, new logs replace the oldest logs.

    Following Table 146 on page 1340 describes the database file size capacity:

Table 146: Database File Size Capacity

Devices	Session	Screen	IDP	Content Security	IPsec-VPN	SKY
SRX300, SRX320, SRX340, SRX345, and SRX550M	1.8G	0.18G	0.18G	0.18G	0.06G	0.18G
SRX1500	12G	2.25G	2.25G	2.25G	0.75G	2.25G
SRX4100 and SRX4200	15G	2.25G	2.25G	2.25G	0.75G	2.25G
SRX4600	22.5G	6G	6G	6G	0.75G	2.25G
vSRX Virtual Firewall	1.8G	0.18G	0.18G	0.18G	0.06G	0.18G

- **Calculating and displaying the reports that are triggered by CLI**—The reports from the database are received from the CLI as the interface. Using the CLI, you can calculate and display the reporting details.

Table Selection

When you want to generate a report from multiple tables, llmd sorts tables based on timestamp and selects tables as per the requested start-time and stop-time.

For example, there are three tables that is table 1 (1 to 3), table 2 (3 to 5) and table 3 (6 to 8). 1 to 3, 3 to 6, and 6 to 8 denotes time stamp of latest and oldest logs. If you request a report from 4 to 6, Junos OS generates report from table 2 and table 3.

Table Lifetime

You can decide table lifetime by configuring `set security log report table-lifetime` command. Junos OS removes the table after the table identify time exceeds the table lifetime. For example, if you configure table lifetime as 2, and the current date is 26-July-2019, then it means on 24-July-2019 00:00:00 logs are removed.

If you change the date and time manually on a device, the table lifetime changes. For example, if a table identify time is 19-July-2019, and you configure table lifetime as 10, Junos OS should remove the table on 29-July-2019. If you change the device date as 18-July-2019, then the table real lifetime becomes 30-July-2019.

Table Dense Mode

In Junos OS Release 19.4R1, we've upgraded the default storage and search mechanism in the on-box logging database to manage logs. You can now customize log storage and search mechanism results. For example, if you expect fewer traffic logs, you can use the default configuration with a start time and a stop time.

However, if you expect a large number of traffic logs and greater time intervals for which the logs will be generated, then enable dense mode. To enable dense mode, use the `set security log report table-mode dense` configuration command.

Chassis Cluster Scenario

For on-box reporting in a chassis cluster, the logs are stored in the local disk on which the device is processing active traffic. These logs are not synchronized to the chassis cluster peer.

Each node is responsible to store logs when each node is processing active traffic. In case of active/passive mode, only the active node processes the traffic and logs are also stored only in the active node. In case of failover, the new active node is processes the traffic and stores logs in its local disk. In case of active/active mode, each node processes its own traffic and logs are stored in the respective nodes.

SEE ALSO

report

Monitor Reports

IN THIS SECTION

- Threats Monitoring Report | **1342**
- Traffic Monitoring Report | **1350**

On-box reporting offers a comprehensive reporting facility where your security management team can spot a security event when it occurs, immediately access and review pertinent details about the event, and quickly decide appropriate remedial action. The J-Web reporting feature provides one- or two-page reports that are equivalent to a compilation of numerous log entries.

This section contains the following topics:

Threats Monitoring Report

IN THIS SECTION

- Purpose | 1342
- Action | 1342

Purpose

Use the Threats Report to monitor general statistics and activity reports of current threats to the network. You can analyze logging data for threat type, source and destination details, and threat frequency information. The report calculates, displays, and refreshes the statistics, providing graphic presentations of the current state of the network.

Action

To view the Threats Report:

1. Click **Threats Report** in the bottom right of the Dashboard, or select **Monitor>Reports>Threats** in the J-Web user interface. The Threats Report appears.
2. Select one of the following tabs:
   - **Statistics** tab. See Table 4 for a description of the page content.
   - **Activities** tab. See Table 5 for a description of the page content.

Table 147: Statistics Tab Output in the Threats Report

Field	Description
General Statistics Pane	

Table 147: Statistics Tab Output in the Threats Report *(Continued)*

Field	Description

Threat Category	One of the following categories of threats:
	• Traffic
	• IDP
	• Content Security
	  • Antivirus
	  • Antispam
	  • Web Filter—Click the Web filter category to display counters for 39 subcategories.
	  • Content Filter
	• Firewall Event
	• DNS

Severity	Severity level of the threat: • emerg • alert • crit • err • warning • notice • info • debug
Hits in past 24 hours	Number of threats encountered per category in the past 24 hours.
Hits in current hour	Number of threats encountered per category in the last hour.

Table 147: Statistics Tab Output in the Threats Report *(Continued)*

Field	Description
Threat Counts in the Past 24 Hours	
By Severity	Graph representing the number of threats received each hour for the past 24 hours sorted by severity level.
By Category	Graph representing the number of threats received each hour for the past 24 hours sorted by category.
X Axis	Twenty-four hour span with the current hour occupying the right-most column of the display. The graph shifts to the left every hour.
Y Axis	Number of threats encountered. The axis automatically scales based on the number of threats encountered.
Most Recent Threats	
Threat Name	Names of the most recent threats. Depending on the threat category, you can click the threat name to go to a scan engine site for a threat description.

Category	Category of each threat:
	• Traffic
	• IDP
	• Content Security
	  • Antivirus
	  • Antispam
	  • Web Filter
	  • Content Filter
	• Firewall Event
	• DNS

Table 147: Statistics Tab Output in the Threats Report *(Continued)*

Field	Description
Source IP/Port	Source IP address (and port number, if applicable) of the threat.
Destination IP/Port	Destination IP address (and port number, if applicable) of the threat.
Protocol	Protocol name of the threat.
Description	Threat identification based on the category type: • Antivirus—URL • Web filter—category • Content filter—reason • Antispam—sender e-mail
Action	Action taken in response to the threat.
Hit Time	Time the threat occurred.
Threat Trend in past 24 hours	

Table 147: Statistics Tab Output in the Threats Report *(Continued)*

Field	Description

Category	Pie chart graphic representing comparative threat counts by category:
	• Traffic
	• IDP
	• Content Security
	  • Antivirus
	  • Antispam
	  • Web Filter
	  • Content Filter
	• Firewall Event
	• DNS

Web Filter Counters Summary	
Category	Web filter count broken down by up to 39 subcategories. Clicking on the Web filter listing in the General Statistics pane opens the Web Filter Counters Summary pane.
Hits in past 24 hours	Number of threats per subcategory in the last 24 hours.
Hits in current hour	Number of threats per subcategory in the last hour.

Table 148: Activities Tab Output in the Threats Report

Field	Function
Most Recent Virus Hits	
Threat Name	Name of the virus threat. Viruses can be based on services, like Web, FTP, or e-mail, or based on severity level.

Table 148: Activities Tab Output in the Threats Report *(Continued)*

Field	Function
Severity	Severity level of each threat: • emerg • alert • crit • err • warning • notice • info • debug
Source IP/Port	IP address (and port number, if applicable) of the source of the threat.
Destination IP/Port	IP address (and port number, if applicable) of the destination of the threat.
Protocol	Protocol name of the threat.
Description	Threat identification based on the category type: • Antivirus—URL • Web filter—category • Content filter—reason • Antispam—sender e-mail
Action	Action taken in response to the threat.
Last Hit Time	Last time the threat occurred.

Table 148: Activities Tab Output in the Threats Report *(Continued)*

Field	Function
Most Recent Spam E-Mail Senders	
From e-mail	E-mail address that was the source of the spam.
Severity	Severity level of the threat: • emerg • alert • crit • err • warning • notice • info • debug
Source IP	IP address of the source of the threat.
Action	Action taken in response to the threat.
Last Send Time	Last time that the spam e-mail was sent.
Recently Blocked URL Requests	
URL	URL request that was blocked.
Source IP/Port	IP address (and port number, if applicable) of the source.
Destination IP/Port	IP address (and port number, if applicable) of the destination.

Table 148: Activities Tab Output in the Threats Report *(Continued)*

Field	Function
Hits in current hour	Number of threats encountered in the last hour.
Most Recent IDP Attacks	
Attack	
Severity	Severity of each threat: • emerg • alert • crit • err • warning • notice • info • debug
Source IP/Port	IP address (and port number, if applicable) of the source.
Destination IP/Port	IP address (and port number, if applicable) of the destination.
Protocol	Protocol name of the threat.
Action	Action taken in response to the threat.
Last Send Time	Last time the IDP threat was sent.

Traffic Monitoring Report

IN THIS SECTION

- Purpose | 1350
- Action | 1350

Purpose

Monitor network traffic by reviewing reports of flow sessions over the past 24 hours. You can analyze logging data for connection statistics and session usage by a transport protocol.

Action

To view network traffic in the past 24 hours, select **Monitor>Reports>Traffic** in the J-Web user interface. See Table 6 for a description of the report.

Table 149: Traffic Report Output

Field	Description
Sessions in Past 24 Hours per Protocol	
Protocol Name	Name of the protocol. To see hourly activity by protocol, click the protocol name and review the “Protocol activities chart” in the lower pane. • TCP • UDP • ICMP
Total Session	Total number of sessions for the protocol in the past 24 hours.
Bytes In (KB)	Total number of incoming bytes in KB.
Bytes Out (KB)	Total number of outgoing bytes in KB.

Table 149: Traffic Report Output *(Continued)*

Field	Description
Packets In	Total number of incoming packets.
Packets Out	Total number of outgoing packets.
Most Recently Closed Sessions	
Source IP/Port	Source IP address (and port number, if applicable) of the closed session.
Destination IP/Port	Destination IP address (and port number, if applicable) of the closed session.
Protocol	Protocol of the closed session. • TCP • UDP • ICMP
Bytes In (KB)	Total number of incoming bytes in KB.
Bytes Out (KB)	Total number of outgoing bytes in KB.
Packets In	Total number of incoming packets.
Packets Out	Total number of outgoing packets.
Timestamp	The time the session was closed.
Protocol Activities Chart	
Bytes In/Out	Graphic representation of traffic as incoming and outgoing bytes per hour. The byte count is for the protocol selected in the Sessions in Past 24 Hours per Protocol pane. Changing the selection causes this chart to refresh immediately.

Table 149: Traffic Report Output *(Continued)*

Field	Description
Packets In/Out	Graphic representation of traffic as incoming and outgoing packets per hour. The packet count is for the protocol selected in the Sessions in Past 24 Hours per Protocol pane. Changing the selection causes this chart to refresh immediately.
Sessions	Graphic representation of traffic as the number of sessions per hour. The session count is for the protocol selected in the Sessions in Past 24 Hours per Protocol pane. Changing the selection causes this chart to refresh immediately.
X Axis	One hour per column for 24 hours.
Y Axis	Byte, packet, or session count.
Protocol Session Chart	
Sessions by Protocol	Graphic representation of the traffic as the current session count per protocol. The protocols displayed are TCP, UDP, and ICMP.

Configure On-Box Binary Security Log Files

SRX Series Firewalls use two types of logs—system logs and security logs—to record system events. System logs record control plane events—for example, when an admin user logs in. Security logs, also known as traffic logs, record data plane events regarding specific traffic handling. For example, Junos OS generates a security log if a security policy denies certain traffic because of a policy violation. For more about system logs, see "Junos OS System Log Overview" on page 1254. For more information about security logs, see "Understanding System Logging for Security Devices" on page 1331.

You can collect and save both system and security logs in binary format either on-box (that is, stored locally on the SRX Series Firewall) or off-box (streamed to a remote device). Using binary format ensures that log files are efficiently stored, which in turn improves CPU utilization.

You can configure security files in binary format using the `log` statement at the `[security]` hierarchy level.

On-box logging is also known as event-mode logging. For stream-mode, off-box security logging, see "Configuring Off-Box Binary Security Log Files" on page 1354. When you configure security logs in

binary format for event-mode logging, you can optionally define the log filename, file path, and other characteristics, as detailed in the following procedure:

1. Specify the logging mode and format for on-box logging.

```
[edit security]
user@host# set log mode event
user@host# set log format binary
```

> **NOTE**: If you configure system logging to send system logs to an external destination (that is, off-box or stream-mode), security logs are also sent to that destination even if you are using event-mode security logging. For more information about sending system logs to an external destination, see "Examples: Configuring System Logging" on page 1286.

> **NOTE**: Off-box and on-box security logging modes cannot be enabled simultaneously.

2. (Optional) Define a name and path for the log file.

> **NOTE**: Security log filename is not mandatory. If security log filename is not configured, by default the bin_messages file is created in the /var/log directory.

```
[edit security]
user@host# set log file name security-binary-log
user@host# set log file path security/log-folder
```

3. (Optional) Change the maximum size of the log file and the maximum number of log files that can be archived.

> **NOTE**: By default, the maximum size of the log file is 3 MB, and a total of three log files can be archived.

In the following sample commands, you set a value of 5 MB and 5 archived files, respectively:

```
[edit security]
user@host# set log file size 5
user@host# set log file files 5
```

4. (Optional) Configure the hpl flag to enable diagnostic traces for the binary security log files. The smf_hpl prefix identifies all binary logging traces.

```
[edit security]
user@host# set log traceoptions flag hpl
```

5. For the default-permit security policy, traffic logs for `RT_FLOW` are generated when a session ends.

```
[edit security]
user@host# set policies from-zone trust to-zone untrust policy default-permit then log
session-close
```

6. (Optional) Traffic logs for `RT_FlOW` are generated when a session starts.

```
[edit security]
user@host# set policies from-zone trust to-zone untrust policy default-permit then log
session-init
```

View the content of the event-mode log file stored on the device using `show security log file` command and use `clear security log file` command to clear the content of the binary event-mode security log file.

NOTE: The `show security log` command displays event-mode security log messages if they are in a text-based format and the `show security log file` command displays event-mode security log messages if they are in binary format (on-box). Off-box binary logging is read by Juniper Secure Analytics (JSA).

Configure Off-Box Binary Security Log Files

SRX Series Firewalls have two types of log: system logs and security logs. System logs record control plane events, for example admin login to the device. For more about system logs, please see "Junos OS

System Log Overview" on page 1254. Security logs, also known as traffic logs, record data plane events regarding specific traffic handling, for example when a security policy denies certain traffic due to some violation of the policy. For more information about security logs, please see "Understanding System Logging for Security Devices" on page 1331.

The two types of log can be collected and saved either on-box or off-box. The procedure below explains how to configure security logs in binary format for off-box (stream-mode) logging.

You can configure security files in binary format using the `log` statement at the `[security]` hierarchy level.

The following procedure specifies binary format for stream-mode security logging, and defines the log filename, path, and log file characteristics. For event-mode, on-box security logging, please see "Configuring On-Box Binary Security Log Files" on page 1352.

1. Specify the logging mode and the format for the log file. For off-box, stream-mode logging:

```
set security log mode stream
set security log stream test-stream format binary host 1.3.54.22
```

NOTE: Off-box and on-box security logging modes cannot be enabled simultaneously.

2. For off-box security logging, specify the source address, which identifies the SRX Series Firewall that generated the log messages. The source address is required.

```
set security log source-address 2.3.45.66
```

3. Optionally, define a log filename and a path.

NOTE: Security log filename is not mandatory. If security log filename is not configured, by default the bin_messages file is created in the /var/log directory.

```
set security log file name security-binary-log
set security log file path security/log-folder
```

4. Optionally, change the maximum size of the log file and the maximum number of log files that can be archived. By default the maximum size of the log file is 3 MB, and a total of three log files can be archived.

```
set security log file size 5
set security log file files 5
```

5. Optionally, select the hpl flag to enable diagnostic traces for binary logging. The prefix smf_hpl identifies all binary logging traces.

```
set security log traceoptions flag hpl
```

6. View the content of the event-mode log file stored on the device using either Juniper Secure Analytics (JSA) or Security Threat Response Manager (STRM).

Configure On-Box Protobuf Security Log Files in Event Mode

Protocol Buffers (Protobuf) is a data format used to serialize structured security logs. You can configure the security log using protobuf format. Data plane use the Protobuf to encode the log and send the log to rtlog process. The rtlog process saves the log file based on the device configuration. By default, the log files are stored in **/var/log/** ***filename.pb*** directory. You can decode the file data using rtlog process.

To configure the Protobuf format in event mode:

1. Specify the logging mode and format for on-box logging.

```
[edit security]
user@host# set log mode event
user@host# set log format protobuf
```

2. Define a name and path for the log file.

```
[edit security]
user@host# set log file name file1.pb
user@host# set log file path /var/tmp
```

3. Change the maximum size of the log file and the maximum number of log files that can be archived.

```
[edit security]
user@host# set log file size 5
user@host# set log file files 5
```

View the content of the protobuf log file stored on the device using the `show security log file file1.pb` command.

`user@host>` **show security log file file1.pb**

```
<14>1 2023-03-17T00:06:55 10.53.78.91 RT_LOG_SELF_TEST - SECINTEL_ACTION_LOG
[junos@2636.1.1.1.2.129 category="secintel" sub-category="CC" action="block" action-
detail="test" http-host="test" threat-severity="5" source-address="1.16.16.16" source-
port="16384" destination-address="2.16.16.16" destination-port="32768" protocol-id="17"
application="test" nested-application="test" feed-name="test" policy-name="test" profile-
name="test" username="Fake username" roles="test" session-id="1" source-zone-name="Fake src
zone" destination-zone-name="Fake dst zone" occur-count="3"]
<14>1 2023-03-17T00:06:55 10.53.78.91 RT_LOG_SELF_TEST - AAMW_ACTION_LOG [junos@2636.1.1.1.2.129
hostname="test" file-category="virus" verdict-number="5" malware-info="Test-File" action="block"
list-hit="test" file-hash-lookup="test" source-address="1.16.16.16" source-port="16384"
destination-address="2.16.16.16" destination-port="32768" protocol-id="17" application="test"
nested-application="test" policy-name="test" username="Fake username" roles="test" session-
id="1" source-zone-name="Fake src zone" destination-zone-name="Fake dst zone" sample-
sha256="da26ba1e13ce4702bd5154789ce1a699ba206c12021d9823380febd795f5b002" file-name="test_name"
url="www.test.com"]
...
```

Configure On-Box Protobuf Security Log Files in Stream Mode

Data plane use the Protobuf to encode the log and send the log to `llmd` process. The `llmd` process saves the log file based on the device configuration. By default, the log files are stored in **/var/traffic-log/*filename.pb*** directory. You can decode the log file data using `uspinfo` process.

To configure the Protobuf format in stream mode to file:

1. Specify the logging mode and format for on-box logging.

```
[edit security]
user@host# set log mode stream
user@host# set log stream s1 format protobuf
```

2. Define a name for the log file.

```
[edit security]
user@host# set log stream s1 file name file2.pb
```

3. Change the maximum size of the log file that can be archived.

```
[edit security]
user@host# set log stream s1 file size 5
```

View the content of the protobuf log file stored on the device using the `show security log stream file file2.pb` command.

`user@host>` **show security log file file2.pb**

```
<14>1 2023-03-15T22:27:34 10.53.78.91 RT_FLOW - RT_FLOW_SESSION_CREATE [junos@2636.1.1.1.2.129
source-address="1.0.0.3" source-port="38800" destination-address="4.0.0.3" destination-port="80"
connection-tag="0" service-name="junos-http" nat-source-address="1.0.0.3" nat-source-
port="38800" nat-destination-address="4.0.0.3" nat-destination-port="80" nat-connection-tag="0"
src-nat-rule-type="N/A" src-nat-rule-name="N/A" dst-nat-rule-type="N/A" dst-nat-rule-name="N/A"
protocol-id="6" policy-name="policy1" source-zone-name="trust" destination-zone-name="untrust"
session-id="69" username="N/A" roles="N/A" packet-incoming-interface="ge-0/0/0.0"
application="HTTP" nested-application="BING" encrypted="No" application-category="Web"
application-sub-category="miscellaneous" application-risk="2" application-characteristics="N/A"
src-vrf-grp="N/A" dst-vrf-grp="N/A" tunnel-inspection="Off" tunnel-inspection-policy-set="root"
source-tenant="N/A" destination-service="N/A"]
<14>1 2023-03-15T22:27:57 10.53.78.91 RT_FLOW - RT_FLOW_SESSION_CLOSE [junos@2636.1.1.1.2.129
reason="TCP FIN" source-address="1.0.0.3" source-port="38800" destination-address="4.0.0.3"
destination-port="80" connection-tag="0" service-name="junos-http" nat-source-address="1.0.0.3"
nat-source-port="38800" nat-destination-address="4.0.0.3" nat-destination-port="80" nat-
connection-tag="0" src-nat-rule-type="N/A" src-nat-rule-name="N/A" dst-nat-rule-type="N/A" dst-
nat-rule-name="N/A" protocol-id="6" policy-name="policy1" source-zone-name="trust" destination-
zone-name="untrust" session-id="69" packets-from-client="11129" bytes-from-client="583566"
packets-from-server="154153" bytes-from-server="218074629" elapsed-time="23" application="HTTP"
```

```
nested-application="BING" username="N/A" roles="N/A" packet-incoming-interface="ge-0/0/0.0"
encrypted="No" application-category="Web" application-sub-category="miscellaneous" application-
risk="2" application-characteristics="N/A" secure-web-proxy-session-type="NA" peer-session-
id="0" peer-source-address="0.0.0.0" peer-source-port="0" peer-destination-address="0.0.0.0"
peer-destination-port="0" hostname="NA NA" src-vrf-grp="N/A" dst-vrf-grp="N/A" tunnel-
inspection="Off" tunnel-inspection-policy-set="root" session-flag="0" source-tenant="N/A"
destination-service="N/A" user-type="N/A"]
...
```

Configure Off-box Protobuf Security Log Files

Data plane use Protobuf format in `stream` and `stream-event` mode to encode the log and send the log to host. The security log data is sent to host using different transport protocol and port number. The host receives the protobuf log and save it to a file. Copy `hplc_collect.py`, `hplc_view.py`, `security_log.xml` and `protobuflog.proto` files to the host. The `hplc_collect.py` is used to collect and save the log files on the host. The `protobuflog.proto` is used to decode the file data on the host and you can view the data using `hplc_view.py`. The files are published to **/share/juniper** and copied to host. The `hplc_collect.py` and `hplc_view.py` files support latest python version 3.

To configure the Protobuf format in stream-event mode to host:

1. Specify the logging mode and log stream format for off-box logging. The `Stream-event` is a combination of stream and event mode.

```
[edit security]
user@host# set log mode stream-event
user@host# set log stream s1 format protobuf
```

2. For off-box security logging, specify the source address, which identifies the SRX Series Firewall that generated the log messages.

```
[edit security]
user@host# set log source-address 10.0.0.3
```

3. Define a name and path for the log file.

```
[edit security]
user@host# set log stream s1 file name proto-1og.pb
user@host# set log file path /var/tmp
```

4. Configure the log stream s1 with host and port settings.

```
[edit security]
user@host# set log stream s1 host 4.0.0.3 port 514
user@host# set log stream s1 transport protocol udp
```

5. Change the maximum size of the log file and the maximum number of log files that can be archived.

```
[edit security]
user@host# set log file size 5
user@host# set log file files 5
```

6. Configure the log.trace file to decode and view the contents of the log.

```
[edit security]
user@host# set log traceoptions file log.trace
```

Send System Log Messages to a File

You can direct system log messages to a file on the CompactFlash (CF) card. The default directory for log files is `/var/log`. To specify a different directory on the CF card, include the complete pathname.

Create a file named `security`, and send log messages of the `authorization` class at the severity level `info` to the file.

To set the filename, the facility, and severity level:

```
{primary:node0}
user@host# set system syslog file security authorization info
```

Configure the System to Send All Log Messages Through eventd

The eventd process of logging configuration is most commonly used for Junos OS. In this configuration, control plane logs and data plane, or security, logs are forwarded from the data plane to the Routing Engine control plane rtlogd process. The rtlogd process then either forwards syslog or sd-syslog-formatted logs to the eventd process or the WELF-formatted logs to the external or remote WELF log collector.

To send all log messages through eventd:

1. Set the eventd process to handle security logs and send them to a remote server.

```
{primary:node0}
user@host# set security log mode event
```

2. Configure the server that will receive the system log messages.

```
{primary:node0}
user@host# set system syslog host hostname any any
```

where *hostname* is the fully qualified hostname or IP address of the server that will receive the logs.

NOTE: To send duplicate logs to a second remote server, repeat the command with a new fully qualified *hostname* or IP address of a second server.

If your deployment is an active/active chassis cluster, you can also configure security logging on the active node to be sent to separate remote servers to achieve logging redundancy.

To rename or redirect one of the logging configurations, you need to delete and recreate it. To delete a configuration:

```
{primary:node0}
user@host# delete security log mode event
```

Release History Table

Release	Description
15.1X49-D100	Starting with Junos OS Release 15.1X49-D100, the default mode for SRX1500 device is stream mode. Prior to Junos OS Release 15.1X49-D100, the default mode for SRX1500 device was event mode.
17.4R2	Starting in Junos OS Release 17.4R2 and later, on SRX300, SRX320, SRX340, SRX345 Series devices and vSRX Virtual Firewall instances, when the device is configured in stream mode, you can configure maximum of eight system log hosts. In Junos OS Release 17.4R2 and earlier releases, you can configure only three system log hosts in the stream mode. If you configure more than three system log hosts, then the following error message is displayed **error: configuration check-out failed**.
Junos OS Release 15.1X49-D100	The on-box reporting feature is enabled by default when you load the factory-default configurations on the SRX Series Firewall with Junos OS Release 15.1X49-D100 or later.
Junos OS Release 19.3R1	Starting in Junos OS Release 19.3R1, SRX300, SRX320, SRX340, SRX345, SRX550, and SRX550M devices default to stream mode.
Junos OS Release 19.3R1	Starting in Junos OS Release 19.3R1 , the factory-default configuration does not include on-box reporting configuration to increase the solid-state drive (SSD) lifetime.

Configure Syslog over TLS

SUMMARY

Learn how to configure your device to transport system log messages (also known as syslog messages) securely over the Transport Layer Security (TLS) protocol.

IN THIS SECTION

- Control Plane Logs | **1363**
- Data Plane Logs | **1368**

Control Plane Logs

IN THIS SECTION

- Example: Configure Syslog over TLS | 1363

Control plane logs, also called system logs, include events that occur on the routing platform. The system sends control plane events to the eventd process on the Routing Engine, which then handles the events by using Junos OS policies, by generating system log messages, or by doing both. You can choose to send control plane logs to a file, user terminal, routing platform console, or remote machine. To generate control plane logs, use the `syslog` statement at the `[system]` hierarchy level.

Example: Configure Syslog over TLS

IN THIS SECTION

- Requirements | 1363
- Overview | 1364
- Configuration | 1364

This example shows how to configure a Juniper Networks device to transport syslog messages (control plane logs) securely over TLS.

Requirements

This example uses the following hardware and software components:

- Junos OS Release 21.2R1 or later
- Junos OS Evolved Release 23.4R1 or later
- Device running Junos OS or Junos OS Evolved (syslog client)
- Syslog server

Overview

You use the TLS protocol to enable secure transportation of system log messages (control plane logs) from the syslog client to the syslog server. TLS uses certificates to authenticate and encrypt the communication.

- Server authentication (or one-way TLS)—Client verifies the identify of the server and trusts the server.
- Mutual authentication—Both the server and client trust each other.

You can choose either server authentication or mutual authentication depending on your network. To quickly access the information you need, click the links in Table 1 on page 1364.

Table 150: TLS Authentication Modes

Authentication Mode	Procedure	Section Where the Information Is Located
Server authentication	Configure PKI Configure the device	"Server Authentication" on page 1366

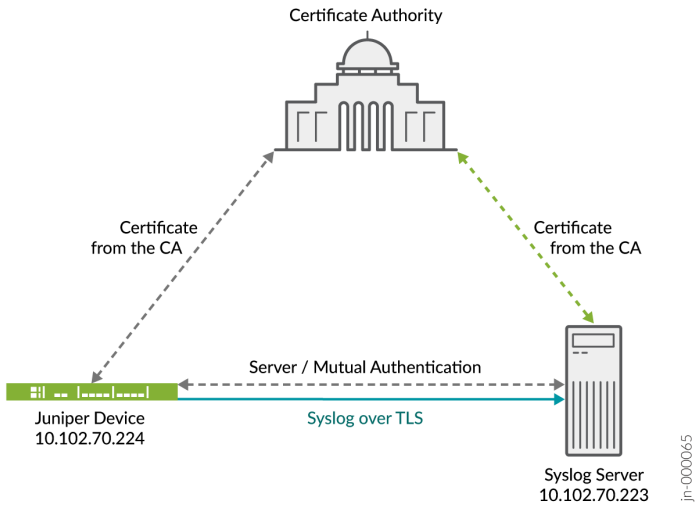
Configuration

IN THIS SECTION

- Public Key Infrastructure (PKI) Configuration Overview | **1365**
- Configure Server Authentication on Your Device | **1366**
- Results | **1367**
- Verification | **1367**

In the following example, we use the TLS protocol to securely transport syslog messages (control plane logs) from the Juniper device to the remote syslog server. Figure 1 shows the basic topology used in this example.

Figure 46: Syslog over TLS



Public Key Infrastructure (PKI) Configuration Overview

To configure PKI on the device:

1. Create a certification authority (CA) profile, and associate a CA identifier with the CA profile. See Example: Configuring a CA Profile.
2. (Optional) Create a revocation check to specify a method for validating the certificate. You can use either certificate revocation lists (CRLs) or the Online Certificate Status Protocol (OCSP). See Certificate Revocation.
3. (Optional) Create a trusted CA group, and add the CA profile to the trusted group. See Configuring a Trusted CA Group.
4. Load the CA certificate on the device. You can load the certificate manually. See Example: Loading CA and Local Certificates Manually. Based on your deployment environment, you can use either Certificate Management Protocol version 2 (CMPv2) or Simple Certificate Enrollment Protocol (SCEP) for online certificate enrollment. See Enrolling a CA Certificate Online Using SCEP and Understanding Certificate Enrollment with CMPv2.
5. (Optional for mutual authentication) Load the local certificate on the device. You can load the local certificate manually. Based on your deployment environment, you can use either CMPv2 or SCEP for online certificate enrollment. See Enrolling a Local Certificate Online Using SCEP and Understanding Certificate Enrollment with CMPv2.

6. Verify that the certificates are loaded successfully. Use the request security pki ca-certificate verify command to check whether the CA certificate has loaded successfully. Use the request security pki local-certificate verify command to verify that the local certificate has loaded successfully.

Configure Server Authentication on Your Device

Step-by-Step Procedure

The following procedure requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see Using the CLI Editor in Configuration Mode.

To configure the device:

1. Specify the syslog server that receives the system log messages. You can specify the IP address of the syslog server or a fully qualified hostname. In this example, use 10.102.70.233 as the IP address of the syslog server.

```
[edit]
user@host# set system syslog host 10.102.70.233 any any
```

2. Specify the port number of the syslog server.

```
[edit]
user@host# set system syslog host 10.102.70.233 port 10514
```

3. Specify the syslog transport protocol for the device. In this example, use TLS as the transport protocol.

```
[edit]
user@host# set system syslog host 10.102.70.223 transport tls
```

4. Specify the name of the trusted certificate authority (CA) group or specify the name of the CA profile to be used. In this example, use example-ca as the CA profile.

```
[edit]
user@host# set system syslog host 10.102.70.223 tlsdetails trusted-ca-group ca-profiles
example-ca
```

5. Configure the device to send all log messages.

```
[edit]
user@host# set system syslog file messages any any
```

6. Commit the configuration.

```
[edit]
user@host# commit
```

Results

In configuration mode, confirm your configuration by using the `show system syslog` command.

```
[edit]
user@host# run show system syslog
host 10.102.70.223
{
    port 10514;
    transport tls;
    tlsdetails {
       local-certificate example-cert;
       trusted-ca-group trusted-ca-group-name {
            ca-profiles example-ca};
      }

    }
}
```

Verification

To verify that the configuration is working properly, enter the `show log` command on the syslog server.

SEE ALSO

tlsdetails

Data Plane Logs

IN THIS SECTION

- Example: Configure the TLS Syslog Protocol on SRX Series Firewalls | 1368

Data plane logs, also called security logs, include security events that are handled inside the data plane. Security logs can be in text or binary format, and you can save them locally (event mode) or configure your device to send the logs to an external server (stream mode). You require binary format for stream mode. We recommend binary format to conserve log space in event mode.

Example: Configure the TLS Syslog Protocol on SRX Series Firewalls

IN THIS SECTION

- Requirements | 1368
- Overview | 1368
- Configuration | 1369
- Verification | 1372

This example shows how to configure the Transport Layer Security (TLS) syslog protocol on SRX Series Firewalls to receive encrypted syslog events from network devices that support TLS syslog event forwarding.

Requirements

Before you begin, enable server certificate verification and encryption or decryption capabilities.

Overview

The TLS syslog protocol enables a log source to receive encrypted syslog events from network devices that support TLS syslog event forwarding. The log source creates a listen port for incoming TLS syslog events and generates a certificate file for the network devices.

In this example, you configure a syslog collector associated with one SSL-I profile. Each SSL-I profile enables the user to specify things such as preferred ciphers suite and trusted CA certificates. You can configure multiple SSL-I profiles and associate the profiles with different collector servers.

Configuration

IN THIS SECTION

- Procedure | 1369

Procedure

CLI Quick Configuration

To quickly configure this section of the example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set security log mode stream
set security log format sd-syslog
set security log source-interface ge-0/0/1.0
set security log transport protocol tls
set security log transport tls-profile ssl-i-tls
set security log stream server1 format sd-syslog
set security log stream server1 category all
set security log stream server1 host 192.0.2.100
set services ssl initiation profile ssl-i-tls protocol-version all
set services ssl initiation profile ssl-i-tls trusted-ca all
set services ssl initiation profile ssl-i-tls actions ignore-server-auth-failure
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see Using the CLI Editor in Configuration Mode in the CLI User Guide.

To configure the TLS syslog protocol:

1. Set the log mode to stream.

```
[edit security]
user@host# set log mode stream
```

2. Specify the structured system log (sd-syslog) format for remote security message logging .

```
[edit security]
user@host# set log format sd-syslog
```

3. Set the host source interface number.

```
[edit security]
user@host# set log source-interface ge-0/0/1.0
```

4. Specify TLS as the security log transport protocol to be used to log the data.

```
[edit security]
user@host# set log transport protocol tls
```

5. Specify the TLS profile name.

```
[edit security]
user@host# set log transport tls-profile ssl-i-tls
```

6. Set the log stream to use the structured syslog format for sending logs to server 1.

```
[edit security]
user@host# set log stream server1 format sd-syslog
```

7. Set the category of server 1 logging to all.

```
[edit security]
user@host# set log stream server1 category all
```

8. Specify server host parameters by entering the server name or IP address.

```
[edit security]
user@host# set log stream server1 host 192.0.2.100
```

9. Define the protocol version *all* for the SSL initiation access profile.

```
[edit services]
user@host# set ssl initiation profile ssl-i-tls protocol-version all
```

10. Attach all CA profile groups to the SSL initiation profile to use when requesting a certificate from the peer.

```
[edit services]
user@host# set ssl initiation profile ssl-i-tls trusted-ca all
```

11. Configure the SSL initiation access profile to ignore the server authentication failure.

```
[edit services]
user@host# set ssl initiation profile ssl-i-tls actions ignore-server-auth-failure
```

Results

In configuration mode, verify your configuration by using the `show security log` command. If the output does not display the intended configuration, then repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# show security log
    mode stream;
        format sd-syslog;
        source-interface ge-0/0/1.0;
        transport {
            protocol tls;
            tls-profile ssl-i-tls;
        }
        stream server1 {
            format sd-syslog;
```

```
            category all;
            host {
            192.0.2.100;
            }
        }
    }
```

```
[edit]
user@host# run show configuration services ssl initiation
    profile ssl-i-tls {
        protocol-version all;
        trusted-ca all;
        actions {
            ignore-server-auth-failure;
        }
    }
```

If you are done configuring the device, enter `commit` from configuration mode.

Verification

To verify that the configuration is working properly, enter the `show log` command on the syslog server.

Monitor Log Messages

IN THIS SECTION

- Monitor System Log Messages | **1373**

Monitor System Log Messages

IN THIS SECTION

- Purpose | 1373
- Action | 1373
- Meaning | 1374

Purpose

Display system log messages about the QFX Series. By looking through a system log file for any entries pertaining to the interface that you are interested in, you can further investigate a problem with an interface on the switch.

Action

To view system log messages:

```
user@switch1> show log messages
```

Sample Output

command-name

```
Nov  4 11:30:01 switch1 newsyslog[2283]: logfile turned over due to size>128K
Nov  4 11:30:01 switch1 newsyslog[2283]: logfile turned over due to size>128K
Nov  4 11:30:06  switch1 chassism[952]: CM ENV Monitor: set fan speed is 65 percent for Fan 1
Nov  4 11:30:06  switch1 chassism[952]: CM ENV Monitor: set fan speed is 65 percent for Fan 2
Nov  4 11:30:06  switch1 chassism[952]: CM ENV Monitor: set fan speed is 65 percent for Fan 3
...
Nov  4 11:52:53  switch1 snmpd[944]: SNMPD_HEALTH_MON_INSTANCE: Health Monitor: jroute daemon
memory usage (Management process): new instance detected (variable:
sysApplElmtRunMemory.5.6.2293)
Nov  4 11:52:53  switch1 snmpd[944]: SNMPD_HEALTH_MON_INSTANCE: Health Monitor: jroute daemon
```

```
memory usage (Command-line interface): new instance detected (variable:
sysApplElmtRunMemory.5.8.2292)
...
Nov  4 12:10:24  switch1 mgd[2293]: UI_CMDLINE_READ_LINE: User 'jsmith', command 'exit '
Nov  4 12:10:27  switch1 mgd[2293]: UI_DBASE_LOGOUT_EVENT: User 'jsmith' exiting configuration
mode
Nov  4 12:10:31  switch1 mgd[2293]: UI_CMDLINE_READ_LINE: User 'jsmith', command 'show log
messages
```

Meaning

The sample output shows the following entries in the **messages** file:

- A new log file was created when the previous file reached the maximum size of 128 kilobytes (KB).
- The fan speed for Fan 1, 2, and 3 is set at 65 percent.
- Health monitoring activity is detected.
- CLI commands were entered by the user jsmith.

SEE ALSO

> *clear log*
>
> *show log*
>
> syslog

PART 11

Network Management and Troubleshooting

Monitoring and Troubleshooting | 1376

Troubleshooting System Performance with Resource Monitoring Methodology | 1426

Configuring Data Path Debugging and Trace Options | 1437

Using MPLS to Diagnose LSPs, VPNs, and Layer 2 Circuits | 1455

Using Packet Capture to Analyze Network Traffic | 1459

Troubleshooting Security Devices | 1488

Monitoring and Troubleshooting

SUMMARY

This section describes the network monitoring and troubleshooting features of Junos OS.

IN THIS SECTION

- Ping Hosts | **1376**
- Monitor Traffic Through the Router or Switch | **1378**
- Dynamic Ternary Content Addressable Memory Overview | **1382**
- Troubleshooting DNS Name Resolution in Logical System Security Policies (Primary Administrators Only) | **1397**
- Troubleshooting the Link Services Interface | **1399**
- Troubleshooting Security Policies | **1412**
- Log Error Messages used for Troubleshooting ISSU-Related Problems | **1416**

Ping Hosts

IN THIS SECTION

- Purpose | **1376**
- Action | **1377**
- Meaning | **1377**

Purpose

Use the CLI `ping` command to verify that a host can be reached over the network. This command is useful for diagnosing host and network connectivity problems. The device sends a series of Internet Control Message Protocol (ICMP) echo (ping) requests to a specified host and receives ICMP echo responses.

Action

To use the `ping` command to send four requests (ping count) to host3:

```
ping host count number
```

Sample Output

command-name

```
ping host3 count 4
user@switch> ping host3 count 4
PING host3.site.net (192.0.2.111): 56 data bytes
64 bytes from 192.0.2.111: icmp_seq=0 ttl=122 time=0.661 ms
64 bytes from 192.0.2.111: icmp_seq=1 ttl=122 time=0.619 ms
64 bytes from 192.0.2.111: icmp_seq=2 ttl=122 time=0.621 ms
64 bytes from 192.0.2.111: icmp_seq=3 ttl=122 time=0.634 ms

--- host3.site.net ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.619/0.634/0.661/0.017 ms
```

Meaning

- The `ping` results show the following information:
  - Size of the ping response packet (in bytes).
  - IP address of the host from which the response was sent.
  - Sequence number of the ping response packet. You can use this value to match the ping response to the corresponding ping request.
  - Time-to-live (ttl) hop-count value of the ping response packet.
  - Total time between the sending of the ping request packet and the receiving of the ping response packet, in milliseconds. This value is also called round-trip time.
  - Number of ping requests (probes) sent to the host.
  - Number of ping responses received from the host.

- Packet loss percentage.
- Round-trip time statistics: minimum, average, maximum, and standard deviation of the round-trip time.

Monitor Traffic Through the Router or Switch

IN THIS SECTION

- Display Real-Time Statistics About All Interfaces on the Router or Switch | **1378**
- Display Real-Time Statistics About an Interface on the Router or Switch | **1380**

For diagnosing a problem, display real-time statistics about the traffic passing through physical interfaces on the router or switch.

To display real-time statistics about physical interfaces, perform these tasks:

Display Real-Time Statistics About All Interfaces on the Router or Switch

IN THIS SECTION

- Purpose | **1378**
- Action | **1379**
- Meaning | **1379**

Purpose

Display real-time statistics about traffic passing through all interfaces on the router or switch.

Action

To display real-time statistics about traffic passing through all interfaces on the router or switch:

```
user@host> monitor interface traffic
```

Sample Output

command-name

```
user@host> monitor interface traffic
host name                 Seconds: 15                 Time: 12:31:09
Interface    Link  Input packets        (pps)     Output packets        (pps)
 so-1/0/0    Down              0          (0)                 0          (0)
 so-1/1/0    Down              0          (0)                 0          (0)
 so-1/1/1    Down              0          (0)                 0          (0)
 so-1/1/2    Down              0          (0)                 0          (0)
 so-1/1/3    Down              0          (0)                 0          (0)
 t3-1/2/0    Down              0          (0)                 0          (0)
 t3-1/2/1    Down              0          (0)                 0          (0)
 t3-1/2/2    Down              0          (0)                 0          (0)
 t3-1/2/3    Down              0          (0)                 0          (0)
 so-2/0/0      Up         211035           (1)             36778          (0)
 so-2/0/1      Up         192753           (1)             36782          (0)
 so-2/0/2      Up         211020           (1)             36779          (0)
 so-2/0/3      Up         211029           (1)             36776          (0)
 so-2/1/0      Up         189378           (1)             36349          (0)
 so-2/1/1    Down              0          (0)             18747          (0)
 so-2/1/2    Down              0          (0)             16078          (0)
 so-2/1/3      Up              0          (0)             80338          (0)
 at-2/3/0      Up              0          (0)                 0          (0)
 at-2/3/1    Down              0          (0)                 0          (0)
Bytes=b, Clear=c, Delta=d, Packets=p, Quit=q or ESC, Rate=r, Up=^U, Down=^D
```

Meaning

The sample output displays traffic data for active interfaces and the amount that each field has changed since the command started or since the counters were cleared by using the `C` key. In this example, the `monitor interface` command has been running for 15 seconds since the command was issued or since the counters last returned to zero.

Display Real-Time Statistics About an Interface on the Router or Switch

IN THIS SECTION

- Purpose | 1380
- Action | 1380
- Meaning | 1381

Purpose

Display real-time statistics about traffic passing through an interface on the router or switch.

Action

To display traffic passing through an interface on the router or switch, use the following Junos OS CLI operational mode command:

```
user@host> monitor interface  interface-name
```

Sample Output

command-name

```
user@host> monitor interface so-0/0/1
Next='n', Quit='q' or ESC, Freeze='f', Thaw='t', Clear='c', Interface='i'
R1
Interface: so-0/0/1, Enabled, Link is Up
Encapsulation: PPP, Keepalives, Speed: OC3 Traffic statistics:
  Input bytes:                  5856541 (88 bps)
  Output bytes:                 6271468 (96 bps)
  Input packets:                 157629 (0 pps)
  Output packets:                157024 (0 pps)
Encapsulation statistics:
  Input keepalives:               42353
  Output keepalives:              42320
  LCP state: Opened
Error statistics:
```

```
  Input errors:                          0
  Input drops:                           0
  Input framing errors:                  0
  Input runts:                           0
  Input giants:                          0
  Policed discards:                      0
  L3 incompletes:                        0
  L2 channel errors:                     0
  L2 mismatch timeouts:                  0
  Carrier transitions:                   1
  Output errors:                         0
  Output drops:                          0
  Aged packets:                          0
Active alarms : None
Active defects: None
SONET error counts/seconds:
  LOS count                              1
  LOF count                              1
  SEF count                              1
  ES-S                                  77
  SES-S                                 77
SONET statistics:
  BIP-B1                                 0
  BIP-B2                                 0
  REI-L                                  0
  BIP-B3                                 0
  REI-P                                  0
Received SONET overhead:  F1        : 0x00  J0        : 0xZ
```

Meaning

The sample output shows the input and output packets for a particular SONET interface (so-0/0/1). The information can include common interface failures, such as SONET/SDH and T3 alarms, loopbacks detected, and increases in framing errors. For more information, see *Checklist for Tracking Error Conditions*.

To control the output of the command while it is running, use the keys shown in Table 1.

Table 151: Output Control Keys for the monitor interface Command

Action	Key
Display information about the next interface. The `monitor interface` command scrolls through the physical or logical interfaces in the same order that they are displayed by the `show interfaces terse` command.	`N`
Display information about a different interface. The command prompts you for the name of a specific interface.	`I`
Freeze the display, halting the display of updated statistics.	`F`
Thaw the display, resuming the display of updated statistics.	`T`
Clear (zero) the current delta counters since `monitor interface` was started. It does not clear the accumulative counter.	`C`
Stop the `monitor interface` command.	`Q`

See the CLI Explorer for details on using match conditions with the `monitor traffic` command.

Dynamic Ternary Content Addressable Memory Overview

IN THIS SECTION

- Applications using Dynamic TCAM Infrastructure | **1383**
- Features Using TCAM Resource | **1383**
- Monitoring TCAM Resource Usage | **1387**
- Example: Monitoring and Troubleshooting the TCAM Resource | **1388**
- Monitoring and Troubleshooting TCAM Resource in ACX Series Routers | **1395**
- Service Scaling on ACX5048 and ACX5096 Routers | **1397**

In ACX Series routers, Ternary Content Addressable Memory (TCAM) is used by various applications like firewall, connectivity fault management, PTPoE, RFC 2544, etc. The Packet Forwarding Engine (PFE) in ACX Series routers uses TCAM with defined TCAM space limits. The allocation of TCAM resources for various filter applications are statically distributed. This static allocation leads to inefficient utilization of TCAM resources when all the filter applications might not use this TCAM resource simultaneously.

The dynamic allocation of TCAM space in ACX routers efficiently allocates the available TCAM resources for various filter applications. In the dynamic TCAM model, various filter applications (such as inet-firewall, bridge-firewall, cfm-filters, etc.) can optimally utilize the available TCAM resources as and when required. Dynamic TCAM resource allocation is usage driven and is dynamically allocated for filter applications on a need basis. When a filter application no longer uses the TCAM space, the resource is freed and available for use by other applications. This dynamic TCAM model caters to higher scale of TCAM resource utilization based on application's demand.

Applications using Dynamic TCAM Infrastructure

The following filter application categories use the dynamic TCAM infrastructure:

- Firewall filter—All the firewall configurations
- Implicit filter—Routing Engine (RE) demons using filters to achieve its functionality. For example, connectivity fault management, IP MAC validation, etc.
- Dynamic filters—Applications using filters to achieve the functionality at the PFE level. For example, logical interface level fixed classifier, RFC 2544, etc. RE demons will not know about these filters.
- System-init filters—Filters that require entries at the system level or fixed set of entries at router's boot sequence. For example, Layer 2 and Layer 3 control protocol trap, default ARP policer, etc.

> **NOTE**: The System-init filter which has the applications for Layer 2 and Layer 3 control protocols trap is essential for the overall system functionality. The applications in this control group consume a fixed and minimal TCAM space from the overall TCAM space. The system-init filter will not use the dynamic TCAM infrastructure and will be created when the router is initialized during the boot sequence.

Features Using TCAM Resource

Applications using the TCAM resource is termed tcam-app in this document. For example, inet-firewall, bridge-firewall, connectivity fault management, link fault management, and so on are all different tcam-apps.

Table 152 on page 1384 describes the list of tcam-apps that use TCAM resources.

Table 152: Features Using TCAM Resource

TCAM Apps/TCAM Users	Feature/Functionality	TCAM Stage
bd-dtag-validate	Bridge domain dual-tagged validate **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Egress
bd-tpid-swap	Bridge domain vlan-map with swap tpid operation	Egress
cfm-bd-filter	Connectivity fault management implicit bridge-domain filters	Ingress
cfm-filter	Connectivity fault management implicit filters	Ingress
cfm-vpls-filter	Connectivity fault management implicit vpls filters **NOTE**: This feature is supported only on ACX5048 and ACX5096 routers.	Ingress
cfm-vpls-ifl-filter	Connectivity fault management implicit vpls logical interface filters **NOTE**: This feature is supported only on ACX5048 and ACX5096 routers.	Ingress
cos-fc	Logical interface level fixed classifier	Pre-ingress
fw-ccc-in	Circuit cross-connect family ingress firewall	Ingress
fw-family-out	Family level egress firewall	Egress
fw-fbf	Firewall filter-based forwarding	Pre-ingress
fw-fbf-inet6	Firewall filter-based forwarding for inet6 family	Pre-ingress
fw-ifl-in	Logical interface level ingress firewall	Ingress

Table 152: Features Using TCAM Resource *(Continued)*

TCAM Apps/TCAM Users	Feature/Functionality	TCAM Stage
fw-ifl-out	Logical interface level egress firewall	Egress
fw-inet-ftf	Inet family ingress firewall on a forwarding-table	Ingress
fw-inet6-ftf	Inet6 family ingress firewall on a forwarding-table	Ingress
fw-inet-in	Inet family ingress firewall	Ingress
fw-inet-rpf	Inet family ingress firewall on RPF fail check	Ingress
fw-inet6-in	Inet6 family ingress firewall	Ingress
fw-inet6-family-out	Inet6 Family level egress firewall	Egress
fw-inet6-rpf	Inet6 family ingress firewall on a RPF fail check	Ingress
fw-inet-pm	Inet family firewall with port-mirror action **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Ingress
fw-l2-in	Bridge family ingress firewall on Layer 2 interface	Ingress
fw-mpls-in	MPLS family ingress firewall	Ingress
fw-semantics	Firewall sharing semantics for CLI configured firewall	Pre-ingress
fw-vpls-in	VPLS family ingress firewall on VPLS interface	Ingress
ifd-src-mac-fil	Physical interface level source MAC filter	Pre-ingress

Table 152: Features Using TCAM Resource *(Continued)*

TCAM Apps/TCAM Users	Feature/Functionality	TCAM Stage
ifl-statistics-in	Logical level interface statistics at ingress	Ingress
ifl-statistics-out	Logical level interface statistics at egress	Egress
ing-out-iff	Ingress application on behalf of egress family filter for log and syslog	Ingress
ip-mac-val	IP MAC validation	Pre-ingress
ip-mac-val-bcast	IP MAC validation for broadcast	Pre-ingress
ipsec-reverse-fil	Reverse filters for IPsec service **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Ingress
irb-cos-rw	IRB CoS rewrite	Egress
lfm-802.3ah-in	Link fault management (IEEE 802.3ah) at ingress **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Ingress
lfm-802.3ah-out	Link fault management (IEEE 802.3ah) at egress	Egress
lo0-inet-fil	Looback interface inet filter	Ingress
lo0-inet6-fil	Looback interface inet6 filter	Ingress
mac-drop-cnt	Statistics for drops by MAC validate and source MAC filters	Ingress
mrouter-port-in	Multicast router port for snooping	Ingress

Table 152: Features Using TCAM Resource *(Continued)*

TCAM Apps/TCAM Users	Feature/Functionality	TCAM Stage
napt-reverse-fil	Reverse filters for network address port translation (NAPT) service **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Ingress
no-local-switching	Bridge no-local-switching	Ingress
ptpoe	Point-to-Point-Over-the-Ethernet traps **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Ingress
ptpoe-cos-rw	CoS rewrite for PTPoE **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Egress
rfc2544-layer2-in	RFC2544 for Layer 2 service at ingress	Pre-ingress
rfc2544-layer2-out	RFC2544 for Layer 2 service at egress **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Egress
service-filter-in	Service filter at ingress **NOTE**: This feature is not supported on ACX5048 and ACX5096 routers.	Ingress

Monitoring TCAM Resource Usage

You can use the show and clear commands to monitor and troubleshoot dynamic TCAM resource usage.

Table 153 on page 1388 summarizes the command-line interface (CLI) commands you can use to monitor and troubleshoot dynamic TCAM resource usage.

Table 153: Show and Clear Commands to Monitor and Troubleshoot Dynamic TCAM

Task	Command
Display the shared and the related applications for a particular application	show pfe tcam app
Display the TCAM resource usage for an application and stages (egress, ingress, and pre-ingress)	show pfe tcam usage (ACX5448) show pfe filter hw summary
Display the TCAM resource usage errors for applications and stages (egress, ingress, and pre-ingress)	show pfe tcam errors
Clears the TCAM resource usage error statistics for applications and stages (egress, ingress, and pre-ingress)	clear pfe tcam-errors

Example: Monitoring and Troubleshooting the TCAM Resource

This section describes a use case where you can monitor and troubleshoot TCAM resources using show commands. In this use case scenario, you have configured Layer 2 services and the Layer 2 service-related applications are using TCAM resources. The dynamic approach, as shown in this example, gives you the complete flexibility to manage TCAM resources on a need basis.

The service requirement is as follows:

- Each bridge domain has one UNI and one NNI interface
- Each UNI interface has:
  - One logical interface level policer to police the traffic at 10 Mbps.
  - Multifield classifier with four terms to assign forwarding class and loss-priority.
- Each UNI interface configures CFM UP MEP at the level 4.
- Each NNI interface configures CFM DOWN MEP at the level 2

Let us consider a scenario where there are 100 services configured on the router. With this scale, all the applications are configured successfully and the status shows **OK** state.

1. Viewing TCAM resource usage for all stages.

To view the TCAM resource usage for all stages (egress, ingress, and pre-ingress), use the `show pfe tcam usage all-tcam-stages detail` command. On ACX5448 routers, use the `show pfe filter hw summary` command to view the TCAM resource usgae.

```
user@host> show pfe tcam usage all-tcam-stages detail
Slot 0

Tcam Resource Stage: Pre-Ingress
--------------------------------
Free [hw-grps: 3 out of 3]
No dynamic tcam usage

Tcam Resource Stage: Ingress
----------------------------
Free [hw-grps: 2 out of 8]
Group: 11, Mode: SINGLE, Hw grps used: 3, Tcam apps: 2
               Used  Allocated  Available     Errors
Tcam-Entries    800       1024        224          0
Counters        800       1024        224          0
Policers          0       1024       1024          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
cfm-filter                500      500        0          3     OK
cfm-bd-filter             300      300        0          2     OK

Group: 8, Mode: DOUBLE, Hw grps used: 2, Tcam apps: 1
               Used  Allocated  Available     Errors
Tcam-Entries    500        512         12          0
Counters        500       1024        524          0
Policers          0       1024       1024          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
fw-l2-in                  500      500        0          2     OK
  fw-semantics              0        X        X          1     OK
```

```
Group: 14, Mode: SINGLE, Hw grps used: 1, Tcam apps: 1
               Used  Allocated  Available     Errors
Tcam-Entries    200        512        312          0
Counters        200        512        312          0
Policers        100        512        412          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
fw-ifl-in                 200      200      100          1     OK

Tcam Resource Stage: Egress
---------------------------
Free [hw-grps: 3 out of 3]
No dynamic tcam usage
```

2. Configure additional Layer 2 services on the router.

   For example, add 20 more services on the router, thereby increasing the total number of services to 120. After adding more services, you can check the status of the configuration by verifying either the syslog message using the command `show log messages`, or by running the `show pfe tcam errors` command.

   The following is a sample syslog message output showing the TCAM resource shortage for Ethernet-switching family filters for newer configurations by running the `show log messages` CLI command.

```
[Sat Jul 11 16:10:33.794 LOG: Err] ACX Error
(dfw):acx_dfw_check_phy_slice_availability :Insufficient phy slices to accomodate grp:13/
IN_IFF_BRIDGE mode:1/DOUBLE
[Sat Jul 11 16:10:33.794 LOG: Err] ACX Error (dfw):acx_dfw_check_resource_availability :Could
not write filter: f-bridge-ge-0/0/0.103-i, insufficient TCAM resources
[Sat Jul 11 16:10:33.794 LOG: Err] ACX Error
(dfw):acx_dfw_update_filter_in_hw :acx_dfw_check_resource_availability failed for filter:f-
bridge-ge-0/0/0.103-i
[Sat Jul 11 16:10:33.794 LOG: Err] ACX Error (dfw):acx_dfw_create_hw_instance :Status:1005
Could not program dfw(f-bridge-ge-0/0/0.103-i) type(IN_IFF_BRIDGE)! [1005]
[Sat Jul 11 16:10:33.794 LOG: Err] ACX Error (dfw):acx_dfw_bind_shim :[1005] Could not create
dfw(f-bridge-ge-0/0/0.103-i) type(IN_IFF_BRIDGE)
[Sat Jul 11 16:10:33.794 LOG: Err] ACX Error (dfw):acx_dfw_bind :[1000] bind failed for
filter f-bridge-ge-0/0/0.103-i
```

If you use the `show pfe tcam errors all-tcam-stages detail` CLI command to verify the status of the configuration, the output will be as shown below:

```
user@host> show pfe tcam errors all-tcam-stages detail
Slot 0

Tcam Resource Stage: Pre-Ingress
--------------------------------
Free [hw-grps: 3 out of 3]
No dynamic tcam usage

Tcam Resource Stage: Ingress
----------------------------
Free [hw-grps: 2 out of 8]
Group: 11, Mode: SINGLE, Hw grps used: 3, Tcam apps: 2
               Used  Allocated  Available     Errors
Tcam-Entries    960       1024         64          0
Counters        960       1024         64          0
Policers          0       1024       1024          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
cfm-filter                600      600        0          3     OK
cfm-bd-filter             360      360        0          2     OK

Group: 8, Mode: DOUBLE, Hw grps used: 2, Tcam apps: 1
               Used  Allocated  Available     Errors
Tcam-Entries    510        512          2         18
Counters        510       1024        514          0
Policers          0       1024       1024          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
fw-l2-in                  510      510        0          2 FAILED
  fw-semantics              0        X        X          1     OK
```

```
App error statistics:
---------------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
fw-l2-in                   18        0        0          2 FAILED
  fw-semantics              0        X        X          1     OK

Group: 14, Mode: SINGLE, Hw grps used: 1, Tcam apps: 1
               Used  Allocated  Available     Errors
Tcam-Entries    240        512        272          0
Counters        240        512        272          0
Policers        120        512        392          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
fw-ifl-in                 240      240      120          1     OK

Tcam Resource Stage: Egress
---------------------------
Free [hw-grps: 3 out of 3]
No dynamic tcam usage
```

The output indicates that the **fw-l2-in** application is running out of TCAM resources and moves into a FAILED state. Although there are two TCAM slices available at the ingress stage, the **fw-l2-in** application is not able to use the available TCAM space due to its mode (DOUBLE), resulting in resource shortage failure.

3. Fixing the applications that have failed due to the shortage of TCAM resouces.

   The **fw-l2-in** application failed because of adding more number of services on the routers, which resulted in shortage of TCAM resources. Although other applications seems to work fine, it is recommended to deactivate or remove the newly added services so that the **fw-l2-in** application moves to an OK state. After removing or deactivating the newly added services, you need to run the `show pfe tcam usage` and `show pfe tcam error` commands to verify that there are no more applications in failed state.

To view the TCAM resource usage for all stages (egress, ingress, and pre-ingress), use the `show pfe tcam usage all-tcam-stages detail` command. For ACX5448 routers, use the `show pfe filter hw summary` command to to view the TCAM resource usage.

```
user@host> show pfe tcam usage all-tcam-stages detail
Slot 0

Tcam Resource Stage: Pre-Ingress
--------------------------------
Free [hw-grps: 3 out of 3]
No dynamic tcam usage

Tcam Resource Stage: Ingress
----------------------------
Free [hw-grps: 2 out of 8]
Group: 11, Mode: SINGLE, Hw grps used: 3, Tcam apps: 2
               Used  Allocated  Available     Errors
Tcam-Entries    800       1024        224          0
Counters        800       1024        224          0
Policers          0       1024       1024          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
cfm-filter                500      500        0          3     OK
cfm-bd-filter             300      300        0          2     OK

Group: 8, Mode: DOUBLE, Hw grps used: 2, Tcam apps: 1
               Used  Allocated  Available     Errors
Tcam-Entries    500        512         12         18
Counters        500       1024        524          0
Policers          0       1024       1024          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
------------------------------------------------------------------
fw-l2-in                  500      500        0          2     OK
  fw-semantics              0        X        X          1     OK
```

```
Group: 14, Mode: SINGLE, Hw grps used: 1, Tcam apps: 1
               Used  Allocated  Available     Errors
Tcam-Entries    200        512        312          0
Counters        200        512        312          0
Policers        100        512        412          0

App tcam usage:
---------------
App-Name              Entries Counters Policers Precedence  State
  Related-App-Name ..
-------------------------------------------------------------------
fw-ifl-in                 200      200      100          1     OK

Tcam Resource Stage: Egress
---------------------------
Free [hw-grps: 3 out of 3]
No dynamic tcam usage
```

To view TCAM resource usage errors for all stages (egress, ingress, and pre-ingress), use the `show pfe tcam errors all-tcam-stages` command.

```
user@host> show pfe tcam errors all-tcam-stages detail
Slot 0

Tcam Resource Stage: Pre-Ingress
--------------------------------
No tcam usage

Tcam Resource Stage: Ingress
----------------------------
Group: 11, Mode: SINGLE, Hw grps used: 3, Tcam apps: 2
               Errors   Resource-Shortage
Tcam-Entries        0                   0
Counters            0                   0
Policers            0                   0

Group: 8, Mode: DOUBLE, Hw grps used: 2, Tcam apps: 1
               Errors   Resource-Shortage
Tcam-Entries       18                   0
Counters            0                   0
Policers            0                   0
```

```
Group: 14, Mode: SINGLE, Hw grps used: 1, Tcam apps: 1
               Errors   Resource-Shortage
Tcam-Entries        0                  0
Counters            0                  0
Policers            0                  0

Tcam Resource Stage: Egress
--------------------------
No tcam usage
```

You can see that all the applications using the TCAM resources are in **OK** state and indicates that the hardware has been successfully configured.

NOTE: As shown in the example, you will need to run the `show pfe tcam errors` and `show pfe tcam usage` commands at each step to ensure that your configurations are valid and that the applications using TCAM resource are in OK state. For ACX5448 routers, use the `show pfe filter hw summary` command to view the TCAM resource usage.

Monitoring and Troubleshooting TCAM Resource in ACX Series Routers

The dynamic allocation of Ternary Content Addressable Memory (TCAM) space in ACX Series efficiently allocates the available TCAM resources for various filter applications. In the dynamic TCAM model, various filter applications (such as inet-firewall, bridge-firewall, cfm-filters, etc.) can optimally utilize the available TCAM resources as and when required. Dynamic TCAM resource allocation is usage driven and is dynamically allocated for filter applications on a need basis. When a filter application no longer uses the TCAM space, the resource is freed and available for use by other applications. This dynamic TCAM model caters to higher scale of TCAM resource utilization based on application's demand. You can use the show and clear commands to monitor and troubleshoot dynamic TCAM resource usage in ACX Series routers.

NOTE: Applications using the TCAM resource is termed tcam-app in this document.

Dynamic Ternary Content Addressable Memory Overview shows the task and the commands to monitor and troubleshoot TCAM resources in ACX Series routers

Table 154: Commands to Monitor and Troubleshoot TCAM Resource in ACX Series

How to	Command
View the shared and the related applications for a particular application.	`show pfe tcam app (list-shared-apps | list-related-apps)`
View the number of applications across all tcam stages.	`show pfe tcam usage all-tcam-stages`
View the number of applications using the TCAM resource at a specified stage.	`show pfe tcam usage tcam-stage (ingress | egress | pre-egress)`
View the TCAM resource used by an application in detail.	`show pfe tcam usage app <application-name> detail`
View the TCAM resource used by an application at a specified stage.	`show pfe tcam usage tcam-stage (ingress | egress | pre-egress) app <application-name>`
Know the number of TCAM resource consumed by a tcam-app	`show pfe tcam usage app <application-name>`
View the TCAM resource usage errors for all stages.	`show pfe tcam errors all-tcam-stages detail`
View the TCAM resource usage errors for a stage	`show pfe tcam errors tcam-stage (ingress | egress | pre-egress)`
View the TCAM resource usage errors for an application.	`show pfe tcam errors app <application-name>`
View the TCAM resource usage errors for an application along with its other shared application.	`show pfe tcam errors app <application-name> shared-usage`
Clear the TCAM resource usage error statistics for all stages.	`clear pfe tcam-errors all-tcam-stages`

Table 154: Commands to Monitor and Troubleshoot TCAM Resource in ACX Series *(Continued)*

How to	Command
Clear the TCAM resource usage error statistics for a specified stage	`clear pfe tcam-errors tcam-stage (`*`ingress`* `/` *`egress`* `/` *`pre-egress`*`)`
Clear the TCAM resource usage error statistics for an application.	`clear pfe tcam-errors app <`*`application-name`*`>`

To know more about dynamic TCAM in ACX Series, see "Dynamic Ternary Content Addressable Memory Overview" on page 1382.

Service Scaling on ACX5048 and ACX5096 Routers

On ACX5048 and ACX5096 routers, a typical service (such as ELINE, ELAN and IP VPN) that is deployed might require applications (such as policers, firewall filters, connectivity fault management IEEE 802.1ag, RFC2544) that uses the dynamic TCAM infrastructure.

NOTE: Service applications that uses TCAM resources is limited by the TCAM resource availability. Therefore, the scale of the service depends upon the consumption of the TCAM resource by such applications.

A sample use case for monitoring and troubleshooting service scale in ACX5048 and ACX5096 routers can be found at the "Dynamic Ternary Content Addressable Memory Overview" on page 1382 section.

Troubleshooting DNS Name Resolution in Logical System Security Policies (Primary Administrators Only)

IN THIS SECTION

- Problem | **1398**
- Cause | **1398**
- Solution | **1398**

Problem

Description

The address of a hostname in an address book entry that is used in a security policy might fail to resolve correctly.

Cause

Normally, address book entries that contain dynamic hostnames refresh automatically for SRX Series Firewalls. The TTL field associated with a DNS entry indicates the time after which the entry should be refreshed in the policy cache. Once the TTL value expires, the SRX Series Firewall automatically refreshes the DNS entry for an address book entry.

However, if the SRX Series Firewall is unable to obtain a response from the DNS server (for example, the DNS request or response packet is lost in the network or the DNS server cannot send a response), the address of a hostname in an address book entry might fail to resolve correctly. This can cause traffic to drop as no security policy or session match is found.

Solution

The primary administrator can use the `show security dns-cache` command to display DNS cache information on the SRX Series Firewall. If the DNS cache information needs to be refreshed, the primary administrator can use the `clear security dns-cache` command.

> **NOTE**: These commands are only available to the primary administrator on devices that are configured for logical systems. This command is not available in user logical systems or on devices that are not configured for logical systems.

SEE ALSO

- Understanding Logical Systems Security Policies

Troubleshooting the Link Services Interface

IN THIS SECTION

- Determine Which CoS Components Are Applied to the Constituent Links | 1399
- Determine What Causes Jitter and Latency on the Multilink Bundle | 1402
- Determine If LFI and Load Balancing Are Working Correctly | 1403
- Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device | 1412

To solve configuration problems on a link services interface:

Determine Which CoS Components Are Applied to the Constituent Links

IN THIS SECTION

- Problem | 1399
- Solution | 1399

Problem

Description

You are configuring a multilink bundle, but you also have traffic without MLPPP encapsulation passing through constituent links of the multilink bundle. Do you apply all CoS components to the constituent links, or is applying them to the multilink bundle enough?

Solution

You can apply a scheduler map to the multilink bundle and its constituent links. Although you can apply several CoS components with the scheduler map, configure only the ones that are required. We recommend that you keep the configuration on the constituent links simple to avoid unnecessary delay in transmission.

Table 5 shows the CoS components to be applied on a multilink bundle and its constituent links.

Table 155: CoS Components Applied on Multilink Bundles and Constituent Links

Cos Component	Multilink Bundle	Constituent Links	Explanation
Classifier	Yes	No	CoS classification takes place on the incoming side of the interface, not on the transmitting side, so no classifiers are needed on constituent links.
Forwarding class	Yes	No	Forwarding class is associated with a queue, and the queue is applied to the interface by a scheduler map. The queue assignment is predetermined on the constituent links. All packets from Q2 of the multilink bundle are assigned to Q2 of the constituent link, and packets from all the other queues are queued to Q0 of the constituent link.

Table 155: CoS Components Applied on Multilink Bundles and Constituent Links *(Continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Scheduler map	Yes	Yes	Apply scheduler maps on the multilink bundle and the constituent link as follows: • Transmit rate—Make sure that the relative order of the transmit rate configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. • Scheduler priority—Make sure that the relative order of the scheduler priority configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. • Buffer size—Because all non-LFI packets from the multilink bundle transit on Q0 of the constituent links, make sure that the buffer size on Q0 of the constituent links is large enough. • RED drop profile—Configure a RED drop profile on the multilink bundle only. Configuring the RED drop profile on the constituent links applies a back pressure mechanism that changes the buffer size and introduces variation. Because this behavior might cause fragment drops on the constituent links, make sure to leave the RED drop profile at the default settings on the constituent links.
Shaping rate for a per-unit scheduler or an interface-level scheduler	No	Yes	Because per-unit scheduling is applied only at the end point, apply this shaping rate to the constituent links only. Any configuration applied earlier is overwritten by the constituent link configuration.

Table 155: CoS Components Applied on Multilink Bundles and Constituent Links *(Continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Transmit-rate exact or queue-level shaping	Yes	No	The interface-level shaping applied on the constituent links overrides any shaping on the queue. Thus apply transmit-rate exact shaping on the multilink bundle only.
Rewrite rules	Yes	No	Rewrite bits are copied from the packet into the fragments automatically during fragmentation. Thus what you configure on the multilink bundle is carried on the fragments to the constituent links.
Virtual channel group	Yes	No	Virtual channel groups are identified through firewall filter rules that are applied on packets only before the multilink bundle. Thus you do not need to apply the virtual channel group configuration to the constituent links.

SEE ALSO

- Class of Service User Guide (Security Devices)

Determine What Causes Jitter and Latency on the Multilink Bundle

IN THIS SECTION

- Problem | 1403
- Solution | 1403

Problem

Description

To test jitter and latency, you send three streams of IP packets. All packets have the same IP precedence settings. After configuring LFI and CRTP, the latency increased even over a noncongested link. How can you reduce jitter and latency?

Solution

To reduce jitter and latency, do the following:

1. Make sure that you have configured a shaping rate on each constituent link.
2. Make sure that you have not configured a shaping rate on the link services interface.
3. Make sure that the configured shaping rate value is equal to the physical interface bandwidth.
4. If shaping rates are configured correctly, and jitter still persists, contact the Juniper Networks Technical Assistance Center (JTAC).

Determine If LFI and Load Balancing Are Working Correctly

IN THIS SECTION

- Problem | **1403**
- Solution | **1404**

Problem

Description

In this case, you have a single network that supports multiple services. The network transmits data and delay-sensitive voice traffic. After configuring MLPPP and LFI, make sure that voice packets are transmitted across the network with very little delay and jitter. How can you find out if voice packets are being treated as LFI packets and load balancing is performed correctly?

Solution

When LFI is enabled, data (non-LFI) packets are encapsulated with an MLPPP header and fragmented to packets of a specified size. The delay-sensitive, voice (LFI) packets are PPP-encapsulated and interleaved between data packet fragments. Queuing and load balancing are performed differently for LFI and non-LFI packets.

To verify that LFI is performed correctly, determine that packets are fragmented and encapsulated as configured. After you know whether a packet is treated as an LFI packet or a non-LFI packet, you can confirm whether the load balancing is performed correctly.

`Solution Scenario`—Suppose two Juniper Networks devices, R0 and R1, are connected by a multilink bundle `lsq-0/0/0.0` that aggregates two serial links, `se-1/0/0` and `se-1/0/1`. On R0 and R1, MLPPP and LFI are enabled on the link services interface and the fragmentation threshold is set to 128 bytes.

In this example, we used a packet generator to generate voice and data streams. You can use the packet capture feature to capture and analyze the packets on the incoming interface.

The following two data streams were sent on the multilink bundle:

- 100 data packets of 200 bytes (larger than the fragmentation threshold)
- 500 data packets of 60 bytes (smaller than the fragmentation threshold)

The following two voice streams were sent on the multilink bundle:

- 100 voice packets of 200 bytes from source port 100
- 300 voice packets of 200 bytes from source port 200

To confirm that LFI and load balancing are performed correctly:

> **NOTE**: Only the significant portions of command output are displayed and described in this example.

1. Verify packet fragmentation. From operational mode, enter the `show interfaces lsq-0/0/0` command to check that large packets are fragmented correctly.

```
user@R0#> show interfaces lsq-0/0/0
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
```

```
  Last flapped   : 2006-08-01 10:45:13 PDT (2w0d 06:06 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface lsq-0/0/0.0 (Index 69) (SNMP ifIndex 42)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
    Bandwidth: 16mbps
    Statistics         Frames      fps       Bytes        bps
    Bundle:
      Fragments:
        Input :             0        0           0          0
        Output:          1100        0      118800          0
      Packets:
        Input :             0        0           0          0
        Output:          1000        0      112000          0
   ...
    Protocol inet, MTU: 1500
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 9.9.9/24, Local: 9.9.9.10
```

Meaning—The output shows a summary of packets transiting the device on the multilink bundle. Verify the following information on the multilink bundle:

- The total number of transiting packets = 1000
- The total number of transiting fragments=1100
- The number of data packets that were fragmented =100

The total number of packets sent (600 + 400) on the multilink bundle match the number of transiting packets (1000), indicating that no packets were dropped.

The number of transiting fragments exceeds the number of transiting packets by 100, indicating that 100 large data packets were correctly fragmented.

Corrective Action—If the packets are not fragmented correctly, check your fragmentation threshold configuration. Packets smaller than the specified fragmentation threshold are not fragmented.

2. Verify packet encapsulation. To find out whether a packet is treated as an LFI or non-LFI packet, determine its encapsulation type. LFI packets are PPP encapsulated, and non-LFI packets are encapsulated with both PPP and MLPPP. PPP and MLPPP encapsulations have different overheads resulting in different-sized packets. You can compare packet sizes to determine the encapsulation type.

A small unfragmented data packet contains a PPP header and a single MLPPP header. In a large fragmented data packet, the first fragment contains a PPP header and an MLPPP header, but the consecutive fragments contain only an MLPPP header.

PPP and MLPPP encapsulations add the following number of bytes to a packet:

- PPP encapsulation adds 7 bytes:

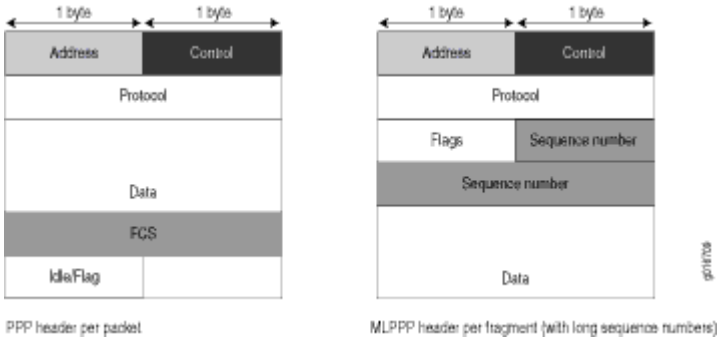
  4 bytes of header+2 bytes of frame check sequence (FCS)+1 byte that is idle or contains a flag

- MLPPP encapsulation adds between 6 and 8 bytes:

  4 bytes of PPP header+2 to 4 bytes of multilink header

Figure 1 shows the overhead added to PPP and MLPPP headers.

Figure 47: PPP and MLPPP Headers



For CRTP packets, the encapsulation overhead and packet size are even smaller than for an LFI packet. For more information, see Example: Configuring the Compressed Real-Time Transport Protocol.

Table 6 shows the encapsulation overhead for a data packet and a voice packet of 70 bytes each. After encapsulation, the size of the data packet is larger than the size of the voice packet.

Table 156: PPP and MLPPP Encapsulation Overhead

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Voice packet (LFI)	PPP	70 bytes	4 + 2 + 1 = 7 bytes	77 bytes

Table 156: PPP and MLPPP Encapsulation Overhead *(Continued)*

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Data fragment (non-LFI) with short sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 2 = 13 bytes	83 bytes
Data fragment (non-LFI) with long sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 4 = 15 bytes	85 bytes

From operational mode, enter the `show interfaces queue` command to display the size of transmitted packet on each queue. Divide the number of bytes transmitted by the number of packets to obtain the size of the packets and determine the encapsulation type.

3. Verify load balancing. From operational mode, enter the `show interfaces queue` command on the multilink bundle and its constituent links to confirm whether load balancing is performed accordingly on the packets.

```
user@R0> show interfaces queue lsq-0/0/0
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :               600                 0 pps
    Bytes                :             44800                 0 bps
  Transmitted:
    Packets              :               600                 0 pps
    Bytes                :             44800                 0 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
```

```
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                400                0 pps
    Bytes                :              61344                0 bps
  Transmitted:
    Packets              :                400                0 pps
    Bytes                :              61344                0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                  0                0 pps
    Bytes                :                  0                0 bps
 …
```

```
user@R0> show interfaces queue se-1/0/0
Physical interface: se-1/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 35
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                350                0 pps
    Bytes                :              24350                0 bps
  Transmitted:
    Packets              :                350                0 pps
    Bytes                :              24350                0 bps
   ...
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                  0                0 pps
    Bytes                :                  0                0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                100                0 pps
    Bytes                :              15272                0 bps
  Transmitted:
    Packets              :                100                0 pps
    Bytes                :              15272                0 bps
    …
```

```
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                 19                0 pps
    Bytes                :                247                0 bps
  Transmitted:
    Packets              :                 19                0 pps
    Bytes                :                247                0 bps
    …
```

```
user@R0> show interfaces queue se-1/0/1
Physical interface: se-1/0/1, Enabled, Physical link is Up
  Interface index: 142, SNMP ifIndex: 38
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                350                0 pps
    Bytes                :              24350                0 bps
  Transmitted:
    Packets              :                350                0 pps
    Bytes                :              24350                0 bps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                  0                0 pps
    Bytes                :                  0                0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                300                0 pps
    Bytes                :              45672                0 bps
  Transmitted:
    Packets              :                300                0 pps
    Bytes                :              45672                0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                 18                0 pps
    Bytes                :                234                0 bps
  Transmitted:
```

```
    Packets              :              18              0 pps
    Bytes                :             234              0 bps
```

Meaning—The output from these commands shows the packets transmitted and queued on each queue of the link services interface and its constituent links. Table 7 shows a summary of these values. (Because the number of transmitted packets equaled the number of queued packets on all the links, this table shows only the queued packets.)

Table 157: Number of Packets Transmitted on a Queue

Packets Queued	Bundle lsq-0/0/0.0	Constituent Link se-1/0/0	Constituent Link se-1/0/1	Explanation
Packets on Q0	600	350	350	The total number of packets transiting the constituent links (350+350 = 700) exceeded the number of packets queued (600) on the multilink bundle.
Packets on Q2	400	100	300	The total number of packets transiting the constituent links equaled the number of packets on the bundle.
Packets on Q3	0	19	18	The packets transiting Q3 of the constituent links are for keepalive messages exchanged between constituent links. Thus no packets were counted on Q3 of the bundle.

On the multilink bundle, verify the following:

- The number of packets queued matches the number transmitted. If the numbers match, no packets were dropped. If more packets were queued than were transmitted, packets were dropped because the buffer was too small. The buffer size on the constituent links controls congestion at the output stage. To correct this problem, increase the buffer size on the constituent links.
- The number of packets transiting Q0 (600) matches the number of large and small data packets received (100+500) on the multilink bundle. If the numbers match, all data packets correctly transited Q0.

- The number of packets transiting Q2 on the multilink bundle (400) matches the number of voice packets received on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

On the constituent links, verify the following:

- The total number of packets transiting Q0 (350+350) matches the number of data packets and data fragments (500+200). If the numbers match, all the data packets after fragmentation correctly transited Q0 of the constituent links.

  Packets transited both constituent links, indicating that load balancing was correctly performed on non-LFI packets.

- The total number of packets transiting Q2 (300+100) on constituent links matches the number of voice packets received (400) on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

  LFI packets from source port `100` transited `se-1/0/0`, and LFI packets from source port `200` transited `se-1/0/1`. Thus all LFI (Q2) packets were hashed based on the source port and correctly transited both constituent links.

`Corrective Action`—If the packets transited only one link, take the following steps to resolve the problem:

a. Determine whether the physical link is `up` (operational) or `down` (unavailable). An unavailable link indicates a problem with the PIM, interface port, or physical connection (link-layer errors). If the link is operational, move to the next step.

b. Verify that the classifiers are correctly defined for non-LFI packets. Make sure that non-LFI packets are not configured to be queued to Q2. All packets queued to Q2 are treated as LFI packets.

c. Verify that at least one of the following values is different in the LFI packets: source address, destination address, IP protocol, source port, or destination port. If the same values are configured for all LFI packets, the packets are all hashed to the same flow and transit the same link.

4. Use the results to verify load balancing.

Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device

IN THIS SECTION

- Problem | 1412
- Solution | 1412

Problem

Description

You are configuring a permanent virtual circuit (PVC) between T1, E1, T3, or E3 interfaces on a Juniper Networks device and a third-party device, and packets are being dropped and ping fails.

Solution

If the third-party device does not have the same FRF.12 support as the Juniper Networks device or supports FRF.12 in a different way, the Juniper Networks device interface on the PVC might discard a fragmented packet containing FRF.12 headers and count it as a "Policed Discard."

As a workaround, configure multilink bundles on both peers, and configure fragmentation thresholds on the multilink bundles.

Troubleshooting Security Policies

IN THIS SECTION

- Synchronizing Policies Between Routing Engine and Packet Forwarding Engine | 1413
- Checking a Security Policy Commit Failure | 1414
- Verifying a Security Policy Commit | 1414
- Debugging Policy Lookup | 1415

Synchronizing Policies Between Routing Engine and Packet Forwarding Engine

IN THIS SECTION

- Problem | 1413
- Solution | 1413

Problem

Description

Security policies are stored in the routing engine and the packet forwarding engine. Security policies are pushed from the Routing Engine to the Packet Forwarding Engine when you commit configurations. If the security policies on the Routing Engine are out of sync with the Packet Forwarding Engine, the commit of a configuration fails. Core dump files may be generated if the commit is tried repeatedly. The out of sync can be due to:

- A policy message from Routing Engine to the Packet Forwarding Engine is lost in transit.
- An error with the routing engine, such as a reused policy UID.

Environment

The policies in the Routing Engine and Packet Forwarding Engine must be in sync for the configuration to be committed. However, under certain circumstances, policies in the Routing Engine and the Packet Forwarding Engine might be out of sync, which causes the commit to fail.

Symptoms

When the policy configurations are modified and the policies are out of sync, the following error message displays - `error: Warning: policy might be out of sync between RE and PFE <SPU-name(s)> Please request security policies check/resync.`

Solution

Use the `show security policies checksum` command to display the security policy checksum value and use the `request security policies resync` command to synchronize the configuration of security policies in the Routing Engine and Packet Forwarding Engine, if the security policies are out of sync.

SEE ALSO

show security policies checksum

request security policies check

request security policies resync

Checking a Security Policy Commit Failure

IN THIS SECTION

- Problem | 1414
- Solution | 1414

Problem

Description

Most policy configuration failures occur during a commit or runtime.

Commit failures are reported directly on the CLI when you execute the CLI command **commit-check** in configuration mode. These errors are configuration errors, and you cannot commit the configuration without fixing these errors.

Solution

To fix these errors, do the following:

1. Review your configuration data.
2. Open the file /var/log/nsd_chk_only. This file is overwritten each time you perform a commit check and contains detailed failure information.

Verifying a Security Policy Commit

IN THIS SECTION

- Problem | 1415
- Solution | 1415

Problem

Description

Upon performing a policy configuration commit, if you notice that the system behavior is incorrect, use the following steps to troubleshoot this problem:

Solution

1. Operational **show** Commands—Execute the operational commands for security policies and verify that the information shown in the output is consistent with what you expected. If not, the configuration needs to be changed appropriately.
2. Traceoptions—Set the `traceoptions` command in your policy configuration. The flags under this hierarchy can be selected as per user analysis of the `show` command output. If you cannot determine what flag to use, the flag option `all` can be used to capture all trace logs.

```
user@host# set security policies traceoptions <flag all>
```

You can also configure an optional filename to capture the logs.

```
user@host# set security policies traceoptions <filename>
```

If you specified a filename in the trace options, you can look in the /var/log/<*filename*> for the log file to ascertain if any errors were reported in the file. (If you did not specify a filename, the default filename is eventd.) The error messages indicate the place of failure and the appropriate reason.

After configuring the trace options, you must recommit the configuration change that caused the incorrect system behavior.

Debugging Policy Lookup

IN THIS SECTION

- Problem | **1416**
- Solution | **1416**

Problem

Description

When you have the correct configuration, but some traffic was incorrectly dropped or permitted, you can enable the `lookup` flag in the security policies traceoptions. The `lookup` flag logs the lookup related traces in the trace file.

Solution

```
user@host# set security policies traceoptions <flag lookup>
```

Log Error Messages used for Troubleshooting ISSU-Related Problems

IN THIS SECTION

- Chassisd Process Errors | **1417**
- Understanding Common Error Handling for ISSU | **1417**
- ISSU Support-Related Errors | **1421**
- Initial Validation Checks Failure | **1421**
- Installation-Related Errors | **1423**
- Redundancy Group Failover Errors | **1424**
- Kernel State Synchronization Errors | **1425**

The following problems might occur during an ISSU upgrade. You can identify the errors by using the details in the logs. For detailed information about specific system log messages, see System Log Explorer.

Chassisd Process Errors

IN THIS SECTION

- Problem | 1417
- Solution | 1417

Problem

Description

Errors related to chassisd.

Solution

Use the error messages to understand the issues related to chassisd.

When ISSU starts, a request is sent to chassisd to check whether there are any problems related to the ISSU from a chassis perspective. If there is a problem, a log message is created.

Understanding Common Error Handling for ISSU

IN THIS SECTION

- Problem | 1417
- Solution | 1418

Problem

Description

You might encounter some problems in the course of an ISSU. This section provides details on how to handle them.

Solution

Any errors encountered during an ISSU result in the creation of log messages, and ISSU continues to function without impact to traffic. If reverting to previous versions is required, the event is either logged or the ISSU is halted, so as not to create any mismatched versions on both nodes of the chassis cluster. Table 158 on page 1418 provides some of the common error conditions and the workarounds for them. The sample messages used in the Table 158 on page 1418 are from the SRX1500 device and are also applicable to all supported SRX Series Firewalls.

Table 158: ISSU-Related Errors and Solutions

Error Conditions	Solutions
Attempt to initiate an ISSU when previous instance of an ISSU is already in progress	The following message is displayed: `warning: ISSU in progress` You can abort the current ISSU process, and initiate the ISSU again using the `request chassis cluster in-service-upgrade abort` command.

Reboot failure on the secondary node	No service downtime occurs, because the primary node continues to provide required services. Detailed console messages are displayed requesting that you manually clear existing ISSU states and restore the chassis cluster.
	`error: [Oct 6 12:30:16]: Reboot secondary node failed (error-code: 4.1)`
	`error: [Oct 6 12:30:16]: ISSU Aborted! Backup node maybe in inconsistent state, Please restore backup node`
	`[Oct 6 12:30:16]: ISSU aborted. But, both nodes are in ISSU window.`
	`Please do the following:`
	`1. Rollback the node with the newer image using rollback command`
	`Note: use the 'node' option in the rollback command`
	`otherwise, images on both nodes will be rolled back`
	`2. Make sure that both nodes (will) have the same image`
	`3. Ensure the node with older image is primary for all RGs`
	`4. Abort ISSU on both nodes`
	`5. Reboot the rolled back node`
	Starting with Junos OS Release 17.4R1, the hold timer for the initial reboot of the secondary node during the ISSU process is extended from 15 minutes (900 seconds) to 45 minutes (2700 seconds) in chassis clusters on SRX1500, SRX4100, SRX4200, and SRX4600 devices.

Table 158: ISSU-Related Errors and Solutions *(Continued)*

Error Conditions	Solutions

Secondary node failed to complete the cold synchronization	The primary node times out if the secondary node fails to complete the cold synchronization. Detailed console messages are displayed that you manually clear existing ISSU states and restore the chassis cluster. No service downtime occurs in this scenario.
	
	`[Oct 3 14:00:46]: timeout waiting for secondary node node1 to sync(error-code: 6.1)`
	`Chassis control process started, pid 36707`
	
	`error: [Oct 3 14:00:46]: ISSU Aborted! Backup node has been upgraded, Please restore backup node`
	`[Oct 3 14:00:46]: ISSU aborted. But, both nodes are in ISSU window.`
	`Please do the following:`
	`1. Rollback the node with the newer image using rollback command`
	`Note: use the 'node' option in the rollback command`
	`otherwise, images on both nodes will be rolled back`
	`2. Make sure that both nodes (will) have the same image`
	`3. Ensure the node with older image is primary for all RGs`
	`4. Abort ISSU on both nodes`
	`5. Reboot the rolled back node`

Table 158: ISSU-Related Errors and Solutions *(Continued)*

Error Conditions	Solutions

Failover of newly upgraded secondary failed	No service downtime occurs, because the primary node continues to provide required services. Detailed console messages are displayed requesting that you manually clear existing ISSU states and restore the chassis cluster.
	
	`[Aug 27 15:28:17]: Secondary node0 ready for failover.`
	`[Aug 27 15:28:17]: Failing over all redundancy-groups to node0`
	`ISSU: Preparing for Switchover`
	`error: remote rg1 priority zero, abort failover.`
	`[Aug 27 15:28:17]: failover all RGs to node node0 failed (error-code: 7.1)`
	`error: [Aug 27 15:28:17]: ISSU Aborted!`
	`[Aug 27 15:28:17]: ISSU aborted. But, both nodes are in ISSU window.`
	`Please do the following:`
	`1. Rollback the node with the newer image using rollback command`
	`Note: use the 'node' option in the rollback command`
	`otherwise, images on both nodes will be rolled back`
	`2. Make sure that both nodes (will) have the same image`
	`3. Ensure the node with older image is primary for all RGs`
	`4. Abort ISSU on both nodes`
	`5. Reboot the rolled back node`
	`{primary:node1}`

Upgrade failure on primary	No service downtime occurs, because the secondary node fails over as primary and continues to provide required services.
Reboot failure on primary node	Before the reboot of the primary node, devices being out of the ISSU setup, no ISSU-related error messages are displayed. The following reboot error message is displayed if any other failure is detected: `Reboot failure on Primary node` `Before the reboot of primary node, devices will be out of ISSU setup and no primary node error messages will be displayed.`

ISSU Support-Related Errors

IN THIS SECTION

- Problem | 1421
- Solution | 1421

Problem

Description

Installation failure occurs because of unsupported software and unsupported feature configuration.

Solution

Use the following error messages to understand the compatibility-related problems:

```
WARNING: Current configuration not compatible with /var/tmp/junos-srx5000-11.4X3.2-domestic.tgz
Exiting in-service-upgrade window
Exiting in-service-upgrade window
```

Initial Validation Checks Failure

IN THIS SECTION

- Problem | 1421
- Solution | 1422

Problem

Description

The initial validation checks fail.

Solution

The validation checks fail if the image is not present or if the image file is corrupt. The following error messages are displayed when initial validation checks fail when the image is not present and the ISSU is aborted:

When Image Is Not Present

```
user@host> ...0120914_srx_12q1_major2.2-539764-domestic.tgz reboot
Chassis ISSU Started
Chassis ISSU Started
ISSU: Validating Image
Initiating in-service-upgrade
Initiating in-service-upgrade
Fetching package...
error: File does not exist: /var/tmp/junos-srx1k3k-12.1I20120914_srx_12q1_major2.2-539764-
domestic.tgz
error: Couldn't retrieve package /var/tmp/junos-srx1k3k-12.1I20120914_srx_12q1_major2.2-539764-
domestic.tgz
Exiting in-service-upgrade window
Exiting in-service-upgrade window
Chassis ISSU Aborted
Chassis ISSU Aborted
Chassis ISSU Aborted
ISSU: IDLE
ISSU aborted; exiting ISSU window.
```

When Image File Is Corrupted

If the image file is corrupted, the following output displays:

```
user@host> ...junos-srx1k3k-11.4X9-domestic.tgz_1 reboot
Chassis ISSU Started
node1:
--------------------------------------------------------------------------
Chassis ISSU Started
ISSU: Validating Image
Initiating in-service-upgrade

node1:
--------------------------------------------------------------------------
```

```
Initiating in-service-upgrade
ERROR: Cannot use /var/tmp/junos-srx1k3k-11.4X9-domestic.tgz_1:
gzip: stdin: invalid compressed data--format violated
tar: Child returned status 1
tar: Error exit delayed from previous errors
ERROR: It may have been corrupted during download.
ERROR: Please try again, making sure to use a binary transfer.
Exiting in-service-upgrade window

node1:
--------------------------------------------------------------------------
Exiting in-service-upgrade window
Chassis ISSU Aborted
Chassis ISSU Aborted

node1:
--------------------------------------------------------------------------
Chassis ISSU Aborted
ISSU: IDLE
ISSU aborted; exiting ISSU window.

{primary:node0}
```

The primary node validates the device configuration to ensure that it can be committed using the new software version. If anything goes wrong, the ISSU aborts and error messages are displayed.

Installation-Related Errors

IN THIS SECTION

- Problem | **1423**
- Solution | **1424**

Problem

Description

The install image file does not exist or the remote site is inaccessible.

Solution

Use the following error messages to understand the installation-related problems:

```
error: File does not exist: /var/tmp/junos-srx5000-11.4X3.2-domest
error: Couldn't retrieve package /var/tmp/junos-srx5000-11.4X3.2-domest
```

ISSU downloads the install image as specified in the ISSU command as an argument. The image file can be a local file or located at a remote site. If the file does not exist or the remote site is inaccessible, an error is reported.

Redundancy Group Failover Errors

IN THIS SECTION

- Problem | 1424
- Solution | 1424

Problem

Description

Problem with automatic redundancy group (RG) failure.

Solution

Use the following error messages to understand the problem:

```
failover all RG 1+ groups to node 0
error: Command failed.  None of the redundancy-groups has been failed over.
       Some redundancy-groups on node1 are already in manual failover mode.
       Please execute 'failover reset all' first..
```

Kernel State Synchronization Errors

IN THIS SECTION

- Problem | 1425
- Solution | 1425

Problem

Description

Errors related to ksyncd.

Solution

Use the following error messages to understand the issues related to ksyncd:

```
Failed to get kernel-replication error information from Standby Routing Engine.
mgd_slave_peer_has_errors() returns error at line 4414 in mgd_package_issu.
```

ISSU checks whether there are any ksyncd errors on the secondary node (node 1) and displays the error message if there are any problems and aborts the upgrade.

Release History Table

Release	Description
17.4R1	Starting with Junos OS Release 17.4R1, the hold timer for the initial reboot of the secondary node during the ISSU process is extended from 15 minutes (900 seconds) to 45 minutes (2700 seconds) in chassis clusters on SRX1500, SRX4100, SRX4200, and SRX4600 devices.

Troubleshooting System Performance with Resource Monitoring Methodology

IN THIS SECTION

- Resource Monitoring Usage Computation Overview | 1426
- Diagnosing and Debugging System Performance by Configuring Memory Resource Usage Monitoring on MX Series Routers | 1429
- Troubleshooting the Mismatch of jnxNatObjects Values for MS-DPC and MS-MIC | 1432
- Managed Objects for Ukernel Memory for a Packet Forwarding Engine in an FPC Slot | 1434
- Managed Objects for Packet Forwarding Engine Memory Statistics Data | 1435
- Managed Objects for Next-Hop, Jtree, and Firewall Filter Memory for a Packet Forwarding Engine in an FPC Slot | 1435
- jnxPfeMemoryErrorsTable | 1436
- pfeMemoryErrors | 1437

Resource Monitoring Usage Computation Overview

IN THIS SECTION

- Resource Monitoring and Usage Computation For Trio-Based Line Cards | 1427
- Resource Monitoring and Usage Computation For I-Chip-Based Line Cards | 1427

You can configure the resource monitoring capability using both the CLI and SNMP MIB queries. You can employ this utility to provision sufficient headroom (memory space limits that are set for the application or virtual router) for monitoring the health and operating efficiency of DPCs and MPCs. You can also analyze and view the usage or consumption of memory for the jtree memory type and for contiguous pages, double words, and free memory pages. The jtree memory on all MX Series router Packet Forwarding Engines has two segments: one segment primarily stores routing tables and related information, and the other segment primarily stores firewall-filter-related information. As the allocation

of more memory for routing tables or firewall filters might disrupt the forwarding operations of a Packet Forwarding Engine, the Junos OS CLI displays a warning to restart all affected FPCs when you commit a configuration that includes the memory-enhanced route statement.

The following sections describe the computation equations and the interpretation of the different memory regions for I-chip-based and Trio-based line cards:

Resource Monitoring and Usage Computation For Trio-Based Line Cards

In Trio-based line cards, memory blocks for next-hop and firewall filters are allocated separately. Also, an expansion memory is present, which is used when the allocated memory for next-hop or firewall filter is fully consumed. Both next-hop and firewall filters can allocate memory from the expansion memory. The encapsulation memory region is specific to I-chip-based line cards and it is not applicable to Trio-based line cards. Therefore, for Trio-based line cards, the percentage of free memory space can be interpreted as follows:

% Free (NH) = (1- (Used NH memory + Used Expansion memory) / (Total NH memory+Total Expansion memory)) × 100

% Free (Firewall or Filter) = (1-(Used FW memory+Used Expansion memory) / (Total FW memory+Total Expansion memory)) × 100

Encapsulation memory is I-chip-specific and is not applicable for Trio-based line cards.

% Free (Encap memory) = Not applicable

Resource Monitoring and Usage Computation For I-Chip-Based Line Cards

I-chip-based line cards contain 32 MB of static RAM (SRAM) memory associated with the route lookup block and 16 MB of SRAM memory associated with the output WAN block.

The route-lookup memory is a single pool of 32 MB memory that is divided into two segments of 16 MB each. In a standard configuration, segment 0 is used for NH and prefixes, and segment 1 is used for firewall or filter. This allocation can be modified by using the route-memory-enhanced option at the [edit chassis] hierarchy level. In a general configuration, NH application can be allocated memory from any of the two segments. Therefore, the percentage of free memory for NH is calculated on 32 MB memory. Currently, firewall applications are allotted memory only from segment 1. As a result, the percentage of free memory to be monitored for firewall starts from the available 16 MB memory in segment 1 only.

For I-chip-based line cards, the percentage of free memory space can be interpreted as follows:

```
% Free (NH) = (32-(Used NH memory+Used FW memory+Used Other application)) / 32×100
```

```
% Free (Firewall or Filter)=(16-(Used NH memory+Used FW memory+Used Other application)) / 16×100
```

The memory size for Output WAN (Iwo) SRAM is 16 MB and stores the Layer 2 descriptors that contain the encapsulation information. This entity is a critical resource and needs to be monitored. This memory space is displayed in the output of the show command as “Encap mem”. The percentage of free memory for the encapsulation region is calculated as follows:

% Free (Encapsulation memory) = (16-(Iwo memory used (L2 descriptors +other applications))) / 16×100

The watermark level configured for next-hop memory is also effective for encapsulation memory. Therefore, if the percentage of free memory for encapsulation region falls below the configured watermark, logs are generated.

If the free memory percentage is lower than the free memory watermark of a specific memory type, the following error message is recorded in the syslog:

“Resource Monitor: FPC <slot no> PFE <pfe inst> <“JNH memory” or “FW/ Filter memory”> is below set watermark <configured watermark>”.

You can configure resource-monitoring tracing operations by using the `traceoptions file <filename> flag flag level level size bytes` statement at the `[edit system services resource-monitor]` hierarchy level. By default, messages are written to **/var/log/rsmonlog**. The error logs associated with socket communication failure (between the Routing Engine and the Packet Forwarding Engine) are useful in diagnosing the problems in the communication between the Routing Engine and the Packet Forwarding Engine.

From the Ukern perspective, MPC5E contains only one Packet Forwarding Engine instance. The show chassis fabric plane command output displays the state of fabric plane connections to the Packet Forwarding Engine. Because two Packet Forwarding Engines exist, you notice PFE-0 and PFE-1 in the output.

```
user@host# run show chassis fabric plane
Fabric management PLANE state
Plane 0
  Plane state: ACTIVE
      FPC 0
          PFE 0 :Links ok
          PFE 1 :Links ok
```

Because only one Packet Forwarding Engine instance for MPC5E exists, the output of the show system resource-monitor fpc command displays only one row corresponding to Packet Forwarding Engine instance 0.

```
user@host# run show system resource-monitor fpc
FPC Resource Usage Summary
```

```
Free Heap Mem Watermark          : 20  %
Free NH Mem Watermark            : 20  %
Free Filter Mem Watermark        : 20  %

* - Watermark reached

                    Heap                      ENCAP mem       NH mem          FW mem
   Slot #         % Free    PFE #           % Free       % Free          % Free
        0             94              0                   NA          83              99
```

The configured watermark is retained across GRES and unified ISSU procedures.

Diagnosing and Debugging System Performance by Configuring Memory Resource Usage Monitoring on MX Series Routers

Junos OS supports a resource monitoring capability using both the CLI and SNMP MIB queries. You can employ this utility to provision sufficient headroom (memory space limits that are set for the application or virtual router) for ensuring system stability, especially the health and operating efficiency of I-chip-based line cards and Trio-based FPCs on MX Series routers. When the memory utilization, either the ukernel memory or ASIC memory reaches a certain threshold, the system operations compromise on the health and traffic-handling stability of the line card and such a trade-off on the system performance can be detrimental for supporting live traffic and protocols.

To configure the properties of the memory resource-utilization functionality:

1. Specify that you want to configure the monitoring mechanism for utilization of different memory resource regions.

   ```
   [edit]
   user@host# edit system services resource-monitor
   ```

   This feature is enabled by default and you cannot disable it manually.

2. Specify the high threshold value, exceeding which warnings or error logs are generated, for all the regions of memory, such as heap or ukernel, next-hop and encapsulation, and firewall filter memory.

   ```
   [edit system services resource-monitor]
   user@host# set high-threshold value
   ```

3. Specify the percentage of free memory space used for next-hops to be monitored with a watermark value.

```
[edit system services resource-monitor]
user@host# set free-nh-memory-watermark percentage
```

4. Specify the percentage of free memory space used for ukernel or heap memory to be monitored with a watermark value.

```
[edit system services resource-monitor]
user@host# set free-heap-memory- watermark percentage
```

5. Specify the percentage of free memory space used for firewall and filter memory to be monitored with a watermark value.

```
[edit system services resource-monitor]
user@host# set free-filter-memory-memory- watermark percentage
```

NOTE:
The default value and the configured value of the watermark value for the percentage of free next-hop memory also applies to encapsulation memory. The default watermark values for the percentage of free ukernel or heap memory, next-hop memory, and firewall filter memory are 20 percent.

6. Disable the generation of error log messages when the utilization of memory resources exceeds the threshold or checkpoint levels. By default, messages are written to /var/log/rsmonlog.

```
[edit system services resource-monitor]
user@host# set no-logging
```

7. Define the resource category that you want to monitor and analyze for ensuring system stability, especially the health and operating efficiency of I-chip-based line cards and Trio-based FPCs on MX Series routers. The resource category includes detailed CPU utilization, session rate, and session

count statistics. You use the resource category statistics to understand the extent to which new attack objects or applications affect performance.

```
[edit system services resource-monitor]
user@host# edit resource-category jtree
```

NOTE: The jtree memory on all MX Series router Packet Forwarding Engines has two segments: one segment primarily stores routing tables and related information, and the other segment primarily stores firewall-filter-related information. The Junos OS provides the memory-enhanced statement to reallocate the jtree memory for routes, firewall filters, and Layer 3 VPNs.

8. Configure the type of resource as contiguous pages for which you want to enable the monitoring mechanism to provide sufficient headroom for ensuring effective system performance and traffic-handling capacity. Specify the high and low threshold value, exceeding which warnings or error logs are generated, for the specified type or region of memory, which is contiguous page in this case.

```
[edit system services resource-monitor resource-category jtree]
user@host# set resource-type contiguous-pages high-threshold percentage
user@host# set resource-type contiguous-pages low-threshold percentage
```

9. Configure the type of resource as free double words (dwords) for which you want to enable the monitoring mechanism to provide sufficient headroom for ensuring effective system performance and traffic-handling capacity. Specify the high and low threshold value, exceeding which warnings or error logs are generated, for the specified type or region of memory, which is free dwords in this case.

```
[edit system services resource-monitor resource-category jtree]
user@host# set resource-type free-dwords high-threshold percentage
user@host# set resource-type free-dwords low-threshold percentage
```

10. Configure the type of resource as free memory pages for which you want to enable the monitoring mechanism to provide sufficient headroom for ensuring effective system performance and traffic-

handling capacity. Specify the high and low threshold value, exceeding which warnings or error logs are generated, for the specified type or region of memory, which is free memory pages in this case.

```
[edit system services resource-monitor resource-category jtree]
user@host# set resource-type free-pages high-threshold percentage
user@host# set resource-type free-pages low-threshold percentage
```

11. View the utilization of memory resources on the Packet Forwarding Engines of an FPC by using the `show system resource-monitor fpc` command. The filter memory denotes the filter counter memory used for firewall filter counters. The asterisk (*) displayed next to each of the memory regions denotes the ones for which the configured threshold is being currently exceeded.

```
user@host# run show system resource-monitor fpc
FPC Resource Usage Summary

Free Heap Mem Watermark         : 20  %
Free NH Mem Watermark           : 20  %
Free Filter Mem Watermark       : 20  %

* - Watermark reached

                    Heap                      ENCAP mem       NH mem          FW mem
   Slot #         % Free    PFE #           % Free       % Free          % Free
        0            94             0                    NA          83              99
```

Troubleshooting the Mismatch of jnxNatObjects Values for MS-DPC and MS-MIC

IN THIS SECTION

- Problem | 1433
- Resolution | 1433

Problem

Description

When both MS-DPC and MS-MIC are deployed in a network and the Network Address Translation (NAT) type is configured as `napt-44`, the output of the `snmp mib walk` command for jnxNatObjects displays different values for MS-DPC and MS-MIC.

Resolution

Configure SNMP to Match jnxNatObjects Values for MS-DPC and MS-MIC

To configure SNMP to match jnxNatObjects values for MS-DPC and MS-MIC:

1. Run the `set services service-set` *`service-set-name`* `nat-options snmp-value-match-msmic` configuration mode command. The following configuration example shows how to configure SNMP to match the values for MS-MIC-specific objects in the jnxNatObjects MIB table with the values for MS-DPC objects.

```
[edit]
user@host# set services service-set Mobile nat-options snmp-value-match-msmic
```

2. Issue the `commit` command to confirm the changes.

```
[edit]
user@host# commit
commit complete
```

3. (Optional) Run the `show snmp mib walk jnxNatObjects` command to verify that the values for MS-MIC-specific objects in the jnxNatObjects MIB table match the values for MS-DPC objects. For example, the following output shows that the values for MS-MIC-specific objects and MS-DPC objects match.

```
[edit]
user@host# run show snmp mib walk jnxNatObjects
jnxNatSrcXlatedAddrType.6.77.111.98.105.108.101 = 1
jnxNatSrcPoolType.6.77.111.98.105.108.101 = 13
jnxNatSrcNumPortAvail.6.77.111.98.105.108.101 = 64512
jnxNatSrcNumPortInuse.6.77.111.98.105.108.101 = 0
jnxNatSrcNumAddressAvail.6.77.111.98.105.108.101 = 1
jnxNatSrcNumAddressInUse.6.77.111.98.105.108.101 = 0
```

```
jnxNatSrcNumSessions.6.77.111.98.105.108.101 = 0
jnxNatRuleType.9.77.111.98.105.108.101.58.116.49 = 13
jnxNatRuleTransHits.9.77.111.98.105.108.101.58.116.49 = 0
jnxNatPoolType.6.77.111.98.105.108.101 = 13
jnxNatPoolTransHits.6.77.111.98.105.108.101 = 0
```

NOTE: You can use the `delete services service-set` *`service-set-name`* `nat-options snmp-value-match-msmic` configuration mode command to disable this feature.

SEE ALSO

Configuring Service Rules

snmp-value-match-msmic

Managed Objects for Ukernel Memory for a Packet Forwarding Engine in an FPC Slot

The `jnxPfeMemoryUkernTable`, whose object identifier is `{jnxPfeMemory 1}`, contains the `JnxPfeMemoryUkernEntry` that retrieves the global ukernel or heap memory statistics for the specified Packet Forwarding Engine slot. Each `JnxPfeMemoryUkernEntry`, whose object identifier is `{jnxPfeMemoryUkernTable 1}`, contains the objects listed in the following table. The `jnxPfeMemoryUkernEntry` denotes the memory utilization, such as the total available memory and the percentage of memory used.

Table 159: jnxPfeMemoryUKernTable

Object	Object ID	Description
`jnxPfeMemoryUkernFreePercent`	`jnxPfeMemoryUkernEntry 3`	Denotes the percentage of free Packet Forwarding Engine memory within the ukern heap.

Managed Objects for Packet Forwarding Engine Memory Statistics Data

The jnxPfeMemory table, whose object identifier is {jnxPfeMib 2} contains the objects listed in Table 160 on page 1435

Table 160: jnxPfeMemory Table

Object	Object ID	Description
jnxPfeMemoryUkernTable	jnxPfeMemory 1	Provides global ukern memory statistics for the specified Packet Forwarding Engine slot.
jnxPfeMemoryForwardingTable	jnxPfeMemory 2	Provides global next-hop (for Trio-based line cards) or Jtree (for I-chip-based line cards) memory utilization and firewall filter memory utilization statistics for the specified Packet Forwarding Engine slot.

Managed Objects for Next-Hop, Jtree, and Firewall Filter Memory for a Packet Forwarding Engine in an FPC Slot

The jnxPfeMemoryForwardingTable, whose object identifier is {jnxPfeMemory 2}, contains JnxPfeMemoryForwardingEntry that retrieves the next-hop memory for Trio- based line cards, jtree memory for I-chip-based line cards, and firewall or filter memory statistics for the specified Packet Forwarding Engine slot for both I- chip and Trio-based line cards. Each jnxPfeMemoryForwardingEntry, whose object identifier is {jnxPfeMemoryForwardingTable 1}, contains the objects listed in the following table.

The jnxPfeMemoryForwardingEntry represents the ASIC instance, ASIC memory used, and ASIC free memory. The jtree memory on all MX Series router Packet Forwarding Engines has two segments: one segment primarily stores routing tables and related information, and the other segment primarily stores firewall-filter-related information. As the allocation of more memory for routing tables or firewall filters might disrupt the forwarding operations of a Packet Forwarding Engine, the Junos OS CLI displays a warning to restart all affected FPCs when you commit a configuration that includes the memory-enhanced route statement. The configuration does not become effective until you restart the FPC or DPC (on MX Series routers).

Table 161: jnxPfeMemoryForwardingTable

Object	Object ID	Description
jnxPfeMemoryForwardingChipSlot	jnxPfeMemoryForwardingEntry 1	Indicates the ASIC instance number in the Packet Forwarding Engine complex.
jnxPfeMemoryType	jnxPfeMemoryForwardingEntry 2	Indicates the Packet Forwarding Engine memory type, where nh = 1, fw = 2, encap = 3.
jnxPfeMemoryForwardingPercentFree	jnxPfeMemoryForwardingEntry 3	Indicates the percentage of memory free for each memory type.

jnxPfeMemoryErrorsTable

The Juniper Networks enterprise-specific Packet Forwarding Engine MIB, whose object ID is {jnxPfeMibRoot 1}, supports a new MIB table, jnxPfeMemoryErrorsTable, to display Packet Forwarding Engine memory error counters. The jnxPfeMemoryErrorsTable, whose object identifier is jnxPfeNotification 3, contains the JnxPfeMemoryErrorsEntry. Each JnxPfeMemoryErrorsEntry, whose object identifier is { jnxPfeMemoryErrorsTable 1 }, contains the objects listed in the following table.

Table 162: jnxPfeMemoryErrorsTable

Object	Object ID	Description
jnxPfeFpcSlot	jnxPfeMemoryErrorsEntry 1	Signifies the FPC slot number for this set of PFE notification
jnxPfeSlot	jnxPfeMemoryErrorsEntry 2	Signifies the PFE slot number for this set of errors
jnxPfeParityErrors	jnxPfeMemoryErrorsEntry 3	Signifies the parity error count
jnxPfeEccErrors	jnxPfeMemoryErrorsEntry 4	Signifies the error-checking code (ECC) error count

pfeMemoryErrors

The pfeMemoryErrorsNotificationPrefix, whose object identifier is {jnxPfeNotification 0}, contains the pfeMemoryErrors attribute. The pfeMemoryErrors object, whose identifier is {pfeMemoryErrorsNotificationPrefix 1} contains the jnxPfeParityErrors and jnxPfeEccErrors objects.

Table 163: pfeMemoryErrors

Object	Object ID	Description
pfeMemoryErrors	pfeMemoryErrorsNotificationPrefix 1	A pfeMemoryErrors notification is sent when the value of jnxPfeParityErrors or jnxPfeEccErrors increases.

Configuring Data Path Debugging and Trace Options

IN THIS SECTION

- Understanding Data Path Debugging for SRX Series Devices | **1438**
- Packet Capture from Operational Mode | **1439**
- Understanding Security Debugging Using Trace Options | **1440**
- Understanding Flow Debugging Using Trace Options | **1440**
- Debugging the Data Path (CLI Procedure) | **1440**
- Setting Flow Debugging Trace Options (CLI Procedure) | **1441**
- Setting Security Trace Options (CLI Procedure) | **1442**
- Displaying Log and Trace Files | **1444**
- Displaying Output for Security Trace Options | **1444**
- Displaying Multicast Trace Operations | **1446**
- Displaying a List of Devices | **1447**
- Example: Configuring End-to-End Debugging on SRX Series Device | **1449**

Understanding Data Path Debugging for SRX Series Devices

Data path debugging, or end-to-end debugging, support provides tracing and debugging at multiple processing units along the packet-processing path. The packet filter can be executed with minimal impact to the production system.

If your goal is to collect packet captures, we strongly recommend leveraging the Operational Mode Packet Capture introduced in Junos OS Release 19.3R1. See "Packet Capture from Operational Mode" on page 1439.

On an SRX Series Firewall, a packet goes through series of events involving different components from ingress to egress processing.

With the data path debugging feature, you can trace and debug (capture packets) at different data points along the processing path. The events available in the packet-processing path are: NP ingress, load-balancing thread (LBT), jexec, packet-ordering thread (POT), and NP egress. You can also enable flow module trace if the security flow trace flag for a certain module is set.

At each event, you can specify any of the four actions (count, packet dump, packet summary, and trace). Data path debugging provides filters to define what packets to capture, and only the matched packets are traced. The packet filter can filter out packets based on logical interface, protocol, source IP address prefix, source port, destination IP address prefix, and destination port.

Data path debugging is supported on SRX4600, SRX5400, SRX5600, and SRX5800.

To enable end-to-end debugging, you must perform the following steps:

1. Define the capture file and specify the maximum capture size.
2. Define the packet filter to trace only a certain type of traffic based on the requirement.
3. Define the action profile specifying the location on the processing path from where to capture the packets (for example, LBT or NP ingress).
4. Enable the data path debugging.
5. Capture traffic.
6. Disable data path debugging.
7. View or analyze the report.

The packet-filtering behavior for the port and interface options is as follows:

- The packet filter traces both IPv4 and IPv6 traffic if only **port** is specified.
- The packet filter traces IPv4, IPV6, and non-IP traffic if only **interface** is specified.

Packet Capture from Operational Mode

Data path debugging or end-to-end debugging provides tracing and debugging at multiple processing units along the packet-processing path. Packet capture is one of the data path debug function. You can execute the packet capture from the operational mode with minimal impact to the production system without committing the configurations.

You can capture the packets using filters to define what packets to capture. The packet filter can filter out packets based on logical interface, protocol, source IP address prefix, source port, destination IP address prefix, and destination port. You can modify the file name, file type, file size, and capture size of the packet capture output. You can also extend the filters into two filters, and swap the values of filters.

Packet capture from operational mode is supported on SRX4600, SRX5400, SRX5600, and SRX5800.

To capture packets from the operational mode, you must perform the following steps:

1. From the operational mode, define the packet filter to trace the type of traffic based on your requirement using the `request packet-capture start` CLI command. See *request packet-capture start* for the available packet capture filter options.
2. Capture the required packets.
3. You can use either the `request packet-capture stop` CLI command to stop the packet capture or after collecting the requested number of packets, the packet capturing stops automatically.
4. View or analyze the captured packet data report.

Limitations of capturing packets from the operational mode are:

1. The configuration mode packet capture and the operational mode packet capture cannot coexist.
2. The operational mode packet capture is a one-time operation and the system does not store the history of this command.
3. You should use the operational mode packet capture in low rate of traffic flow.

SEE ALSO

request packet-capture start

request packet-capture stop

Understanding Security Debugging Using Trace Options

The Junos OS trace function allows applications to write security debugging information to a file. The information that appears in this file is based on criteria you set. You can use this information to analyze security application issues.

The trace function operates in a distributed manner, with each thread writing to its own trace buffer. These trace buffers are then collected at one point, sorted, and written to trace files. Trace messages are delivered using the InterProcess Communications (IPC) protocol. A trace message has a lower priority than that of control protocol packets such as BGP, OSPF, and IKE, and therefore delivery is not considered to be as reliable.

Understanding Flow Debugging Using Trace Options

For flow trace options, you can define a packet filter using combinations of **destination-port**, **destination-prefix**, **interface**, **protocol**, **source-port**, and **source-prefix**. If the security flow trace flag for a certain module is set, the packet matching the specific packet filter triggers flow tracing and writes debugging information to the trace file.

Debugging the Data Path (CLI Procedure)

Data path debugging is supported on SRX5400, SRX5600, and SRX5800.

To configure the device for data path debugging:

1. Specify the following request command to set the data path debugging for the multiple processing units along the packet-processing path:

```
[edit]
user@host# set security datapath-debug
```

2. Specify the trace options for data path-debug using the following command:

```
[edit]
user@host# set security datapath-debug traceoptions
```

3. Using the request security packet-filter command, you can set the packet filter to specify the related packets to perform data path-debug action. A maximum of four filters are supported at the same time. For example, the following command sets the first packet-filter:

```
[edit]
user@host# set security datapath-debug packet-filter name
```

4. Using the request security action-profile command, you can set the action for the packet match for a specified filter. Only the default action profile is supported, which is the trace option for network processor ezchip ingress, ezchip egress, spu.lbt, and spu.pot:

```
[edit]
user@host# set security datapath-debug packet-filter name action-profile
```

Setting Flow Debugging Trace Options (CLI Procedure)

The following examples display the options you can set by using `security flow traceoptions`.

- To match the imap destination port for the filter1 packet filter, use the following statement:

```
[edit]
user@host# set security flow traceoptions packet-filter filter1 destination-port imap
```

- To set the 1.2.3.4 destination IPv4 prefix address for the filter1 packet filter, use the following statement:

```
[edit]
user@host# set security flow traceoptions packet-filter filter1 destination-prefix 1.2.3.4
```

- To set the fxp0 logical interface for the filter1 packet filter, use the following statement:

```
[edit]
user@host# set security flow traceoptions packet-filter filter1 interface fxp0
```

- To match the TCP IP protocol for the filter1 packet filter, use the following statement:

```
[edit]
user@host# set security flow traceoptions packet-filter filter1 protocol tcp
```

- To match the HTTP source port for the filter1 packet filter, use the following statement:

```
[edit]
user@host# set security flow traceoptions packet-filter filter1 source-port http
```

- To set the 5.6.7.8 IPv4 prefix address for the filter1 packet filter, use the following statement:

```
[edit]
user@host# set security flow traceoptions packet-filter filter1 source-prefix 5.6.7.8
```

Setting Security Trace Options (CLI Procedure)

Use the following configuration statements to configure security trace options in the CLI configuration editor.

- To disable remote tracing, enter the following statement:

```
[edit]
user@host# set security traceoptions no-remote-trace
```

- To write trace messages to a local file, enter the following statement. The system saves the trace file in the **/var/log/** directory.

```
[edit]
user@host# set security traceoptions use-local-files
```

- To specify a name for the trace file, enter the following statement. Valid values range from 1 and 1024 characters. The name cannot include spaces, /, or % characters. The default filename is security.

```
[edit]
user@host# set security traceoptions file filename
```

- To specify the maximum number of trace files that can accumulate, enter the following statement. Valid values range from 2 to 1000. The default value is 3.

```
[edit]
user@host# set security traceoptions file files 3
```

- To specify the match criteria that you want the system to use when logging information to the file, enter the following statement. Enter a regular expression. Wildcard (*) characters are accepted.

```
[edit]
user@host# set security traceoptions file match *thread
```

- To allow any user to read the trace file, enter the `world-readable` statement. Otherwise, enter the `no-world-readable` statement.

```
[edit]
user@host# set security traceoptions file world-readable
user@host# set security traceoptions file no-world-readable
```

- To specify the maximum size to which the trace file can grow, enter the following statement. Once the file reaches the specified size, it is compressed and renamed *filename*0.gz, the next file is named *filename*1.gz, and so on. Valid values range from 10240 to 1,073,741,824.

```
[edit]
user@host# set security traceoptions file size 10240
```

- To turn on trace options and to perform more than one tracing operation, set the following flags.

```
[edit]
user@host# set security traceoptions flag all
```

```
user@host# set security traceoptions flag compilation
user@host#  set security traceoptions flag configuration
user@host# set security traceoptions flag routing-socket
```

- To specify the groups that these trace option settings do or do not apply to, enter the following statements:

```
[edit]
user@host# set security traceoptions apply-groups value
user@host# set security traceoptions apply-groups-except value
```

Displaying Log and Trace Files

Enter the `monitor start` command to display real-time additions to system logs and trace files:

```
user@host> monitor start filename
```

When the device adds a record to the file specified by *filename*, the record displays on the screen. For example, if you have configured a system log file named `system-log` (by including the `syslog` statement at the [`edit system`] hierarchy level), you can enter the `monitor start system-log` command to display the records added to the system log.

To display a list of files that are being monitored, enter the `monitor list` command. To stop the display of records for a specified file, enter the `monitor stop` *filename* command.

Displaying Output for Security Trace Options

IN THIS SECTION

- Purpose | 1445
- Action | 1445

Purpose

Display output for security trace options.

Action

Use the `show security traceoptions` command to display the output of your trace files. For example:

```
[edit]
user@host # show security traceoptions file usp_trace
user@host # show security traceoptions flag all
user@host # show security traceoptions rate-limit 888
```

The output for this example is as follows:

```
Apr 11 16:06:42 21:13:15.750395:CID-906489336:FPC-01:PIC-01:THREAD_ID-01:PFE:now update
0x3607edf8df8in 0x3607e8d0
Apr 11 16:06:42 21:13:15.874058:CID-1529687608:FPC-01:PIC-01:THREAD_ID-01:CTRL:Enter
Function[util_ssam_handler]
Apr 11 16:06:42 21:13:15.874485:CID-00:FPC-01:PIC-01:THREAD_ID-01:CTRL:default1: Rate limit
changed to 888
Apr 11 16:06:42 21:13:15.874538:CID-00:FPC-01:PIC-01:THREAD_ID-01:CTRL:default1: Destination ID
set to 1
Apr 11 16:06:42 21:13:15.874651:CID-00:FPC-01:PIC-01:THREAD_ID-01:CTRL:default2: Rate limit
changed to 888
Apr 11 16:06:42 21:13:15.874832:CID-00:FPC-01:PIC-01:THREAD_ID-01:CTRL:default2: Destination ID
set to 1
Apr 11 16:06:42 21:13:15.874942:CID-00:FPC-01:PIC-01:THREAD_ID-01:CTRL:default3: Rate limit
changed to 888
Apr 11 16:06:42 21:13:15.874997:CID-00:FPC-01:PIC-01:THREAD_ID-01:CTRL:default3: Destination ID
set to 1
```

Displaying Multicast Trace Operations

To monitor and display multicast trace operations, enter the `mtrace monitor` command:

```
user@host> mtrace monitor
```

```
Mtrace query at Apr 21 16:00:54 by 192.1.30.2, resp to 224.0.1.32, qid 2a83aa packet from
192.1.30.2 to 224.0.0.2 from 192.1.30.2 to 192.1.4.1 via group 224.1.1.1 (mxhop=60)  Mtrace
query at Apr 21 16:00:57 by 192.1.30.2, resp to 224.0.1.32, qid 25dc17 packet from 192.1.30.2 to
224.0.0.2 from 192.1.30.2 to 192.1.4.1 via group 224.1.1.1 (mxhop=60)  Mtrace query at Apr 21
16:01:00 by 192.1.30.2, resp to same, qid 20e046 packet from 192.1.30.2 to 224.0.0.2 from
192.1.30.2 to 192.1.4.1 via group 224.1.1.1 (mxhop=60)  Mtrace query at Apr 21 16:01:10 by
192.1.30.2, resp to same, qid 1d25ad packet from 192.1.30.2 to 224.0.0.2 from 192.1.30.2 to
192.1.4.1 via group 224.1.1.1 (mxhop=60)
```

This example displays only `mtrace` queries. However, when the device captures an `mtrace` response, the display is similar, but the complete `mtrace` response also appears (exactly as it is appears in the `mtrace from-source` command output).

Table 164 on page 1446 summarizes the output fields of the display.

Table 164: CLI mtrace monitor Command Output Summary

Field	Description
Mtrace *operation-type* at *time-of-day*	• *operation-type*—Type of multicast trace operation: `query` or `response`. • *time-of-day*—Date and time the multicast trace query or response was captured.
by	IP address of the host issuing the query.
resp to *address*	*address*—Response destination address.
qid *qid*	*qid*—Query ID number.

Table 164: CLI mtrace monitor Command Output Summary *(Continued)*

Field	Description
packet from *source* to *destination*	• *source*—IP address of the source of the query or response. • *destination*—IP address of the destination of the query or response.
from *source* to *destination*	• *source*—IP address of the multicast source. • *destination*—IP address of the multicast destination.
via group *address*	*address*—Group address being traced.
mxhop=*number*	*number*—Maximum hop setting.

Displaying a List of Devices

To display a list of devices between the device and a specified destination host, enter the `traceroute` command with the following syntax:

```
user@host> traceroute host <interface interface-name> <as-number-lookup> <bypass-routing>
<gateway address> <inet | inet6> <no-resolve> <routing-instance routing-instance-name>
<source source-address> <tos number> <ttl number> <wait seconds>
```

Table 165 on page 1447 describes the `traceroute` command options.

Table 165: CLI traceroute Command Options

Option	Description
host	Sends traceroute packets to the hostname or IP address you specify.
interface *interface-name*	(Optional) Sends the traceroute packets on the interface you specify. If you do not include this option, traceroute packets are sent on all interfaces.

Table 165: CLI traceroute Command Options *(Continued)*

Option	Description
`as-number-lookup`	(Optional) Displays the autonomous system (AS) number of each intermediate hop between the device and the destination host.
`bypass-routing`	(Optional) Bypasses the routing tables and sends the traceroute packets only to hosts on directly attached interfaces. If the host is not on a directly attached interface, an error message is returned. Use this option to display a route to a local system through an interface that has no route through it.
`gateway` *address*	(Optional) Uses the gateway you specify to route through.
`inet`	(Optional) Forces the traceroute packets to an IPv4 destination.
`inet6`	(Optional) Forces the traceroute packets to an IPv6 destination.
`no-resolve`	(Optional) Suppresses the display of the hostnames of the hops along the path.
`routing-instance` *routing-instance-name*	(Optional) Uses the routing instance you specify for the traceroute.
`source` *address*	(Optional) Uses the source address that you specify, in the traceroute packet.
`tos` *number*	(Optional) Sets the type-of-service (TOS) value in the IP header of the traceroute packet. Specify a value from `0` through `255`.
`ttl` *number*	(Optional) Sets the time-to-live (TTL) value for the traceroute packet. Specify a hop count from `0` through `128`.
`wait` *seconds*	(Optional) Sets the maximum time to wait for a response.

To quit the `traceroute` command, press Ctrl-C.

The following is sample output from a `traceroute` command:

```
user@host> traceroute host2
```

```
traceroute to 173.24.232.66 (172.24.230.41), 30 hops max, 40 byte packets  1  173.18.42.253
(173.18.42.253)  0.482 ms  0.346 ms  0.318 ms  2  host4.site1.net (173.18.253.5)  0.401 ms
0.435 ms  0.359 ms  3  host5.site1.net (173.18.253.5)  0.401 ms  0.360 ms  0.357 ms  4
173.24.232.65 (173.24.232.65)  0.420 ms  0.456 ms  0.378 ms  5  173.24.232.66 (173.24.232.66)
0.830 ms  0.779 ms  0.834 ms
```

The fields in the display are the same as those displayed by the J-Web traceroute diagnostic tool.

Example: Configuring End-to-End Debugging on SRX Series Device

IN THIS SECTION

- Requirements | **1449**
- Overview | **1450**
- Configuration | **1450**
- Enabling Data Path Debugging | **1453**
- Verification | **1454**

This example shows how to configure and enable end-to-end debugging on an SRX Series Firewall with an SRX5K-MPC.

Requirements

This example uses the following hardware and software components:

- SRX5600 device with an SRX5K-MPC installed with 100-Gigabit Ethernet CFP transceiver
- Junos OS Release 12.1X47-D15 or later for SRX Series Firewalls

Before you begin:

- See "Understanding Data Path Debugging for SRX Series Devices" on page 1438.

No special configuration beyond device initialization is required before configuring this feature.

Overview

Data path debugging enhances troubleshooting capabilities by providing tracing and debugging at multiple processing units along the packet-processing path. With the data path debugging feature, you can trace and debug (capture packets) at different data points along the processing path. At each event, you can specify an action (count, packet dump, packet summary, and trace) and you can set filters to define what packets to capture.

In this example, you define a traffic filter, and then you apply an action profile. The action profile specifies a variety of actions on the processing unit. The ingress and egress are specified as locations on the processing path to capture the data for incoming and outgoing traffic.

Next, you enable data path debugging in operational mode, and finally you view the data capture report.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Configuration

IN THIS SECTION

- Procedure | 1450

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set security datapath-debug traceoptions file e2e.trace size 10m
set security datapath-debug capture-file e2e.pcap format pcap
set security datapath-debug maximum-capture-size 1500
set security datapath-debug capture-file files 10
set security datapath-debug action-profile profile-1 preserve-trace-order
```

```
set security datapath-debug action-profile profile-1 record-pic-history
set security datapath-debug action-profile profile-1 event np-ingress trace
set security datapath-debug action-profile profile-1 event np-ingress count
set security datapath-debug action-profile profile-1 event np-ingress packet-summary
set security datapath-debug action-profile profile-1 event np-egress trace
set security datapath-debug action-profile profile-1 event np-egress count
set security datapath-debug action-profile profile-1 event np-egress packet-summary
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure data path debugging:

1. Edit the security datapath debugging option for the multiple processing units along the packet-processing path:

```
[edit]
user@host# edit security datapath-debug
```

2. Enable the capture file, file format, file size, and the number of files.

```
[edit security datapath-debug]
user@host# set traceoptions file e2e.trace size 10m
user@host# set capture-file e2e.pcap format pcap;
user@host# set maximum-capture-size 1500
user@host# set capture-file files 10
```

3. Configure action profile, event type, and actions for the action profile.

```
[edit security datapath-debug]
user@host# set action-profile profile-1 preserve-trace-order
user@host# set action-profile profile-1 record-pic-history
user@host# set action-profile profile-1 event np-ingress trace
user@host# set action-profile profile-1 event np-ingress count
user@host# set action-profile profile-1 event np-ingress packet-summary
user@host# set action-profile profile-1 event np-egress trace
```

```
user@host# set action-profile profile-1 event np-egress count
user@host# set action-profile profile-1 event np-egress packet-summary
```

Results

From configuration mode, confirm your configuration by entering the `show security datapath-debug` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
traceoptions {
    file e2e.trace size 10m;
}
capture-file e2e.pcap format pcap;
maximum-capture-size 1500;
capture-file files 10;
action-profile {
    profile-1 {
        preserve-trace-order;
        record-pic-history;
        event np-ingress {
            trace;
            packet-summary;
            packet-dump;
        }
        event np-egress {
            trace;
            packet-summary;
            packet-dump;
        }
    }
}
```

If you are done configuring the device, enter `commit` from configuration mode.

Enabling Data Path Debugging

IN THIS SECTION

- Procedure | 1453

Procedure

Step-by-Step Procedure

After configuring data path debugging, you must start the process on the device from operational mode.

1. Enable data path debugging.

```
user@host> request security datapath-debug capture start
```

```
datapath-debug capture started on file datapcap
```

2. Before you verify the configuration and view the reports, you must disable data path debugging.

```
user@host> request security datapath-debug capture stop
```

```
datapath-debug capture succesfully stopped, use show security datapath-debug capture to view
```

NOTE: You must stop the debug process after you have finished capturing the data. If you attempt to open the captured files without stopping the debug process, the files obtained cannot be opened through any third-party software (for example, tcpdump and wireshark).

Verification

IN THIS SECTION

- Verifying Data Path Debug Packet Capture Details | 1454

Confirm that the configuration is working properly.

Verifying Data Path Debug Packet Capture Details

Purpose

Verify the data captured by enabling the data path debugging configuration.

Action

From operational mode, enter the `show security datapath-debug capture` command.

```
Packet 8, len 152: (C2/F2/P0/SEQ:57935:np-ingress)
00 10 db ff 10 02 00 30 48 83 8d 4f 08 00 45 00
00 54 00 00 40 00 40 01 9f c7 c8 07 05 69 c8 08
05 69 08 00 91 1f 8f 03 2a a2 ae 66 85 53 8c 7d
02 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15
16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
36 37
Packet 9, len 152: (C2/F2/P0/SEQ:57935:np-egress)
00 30 48 8d 1a bf 00 10 db ff 10 03 08 00 45 00
00 54 00 00 40 00 3f 01 a0 c7 c8 07 05 69 c8 08
05 69 08 00 91 1f 8f 03 2a a2 ae 66 85 53 8c 7d
02 00 08 09 0a 0b 0c 0d 0e 0f 10 11 12 13 14 15
16 17 18 19 1a 1b 1c 1d 1e 1f 20 21 22 23 24 25
26 27 28 29 2a 2b 2c 2d 2e 2f 30 31 32 33 34 35
36 37....
```

For brevity, the `show` command output is truncated to display only a few samples. Additional samples have been replaced with ellipses (...).

To view the results, from CLI operational mode, access the local UNIX shell and navigate to the directory **/var/log/<*file-name*>**. The result can be read by using the `tcpdump` utility.

```
user@host>start shell
%tcpdump -nr/var/log/e2e.pcap
```

```
21:50:04.288767 C0/F3 event:1(np-ingress) SEQ:1 IP 192.168.14.2 > 192.168.13.2: ICMP echo
request, id 57627, seq 0, length 64
21:50:04.292590 C0/F3 event:2(np-egress) SEQ:1 IP 192.168.14.2 > 192.168.13.2: ICMP echo
request, id 57627, seq 0, length 64
1:50:04.295164 C0/F3 event:1(np-ingress) SEQ:2 IP 192.168.13.2 > 192.168.14.2: ICMP echo reply,
id 57627, seq 0, length 64
21:50:04.295284 C0/F3 event:2(np-egress) SEQ:2 IP 192.168.13.2 > 192.168.14.2: ICMP echo reply,
id 57627, seq 0, length 64
```

NOTE: If you are finished with troubleshooting the data path debugging, remove all `traceoptions` (not limited to flow traceoptions) and the complete data path debug configuration, including the data path debug configuration for packet capturing (packet-dump), which needs to be started/stopped manually. If any part of the debugging configuration remains active, it will continue to use the resources of the device (CPU/memory).

Using MPLS to Diagnose LSPs, VPNs, and Layer 2 Circuits

IN THIS SECTION

- MPLS Connection Checking Overview | **1456**

MPLS Connection Checking Overview

IN THIS SECTION

- MPLS Enabled | 1458
- Loopback Address | 1459
- Source Address for Probes | 1459
- Using the ping Command | 1459

Use either the J-Web ping MPLS diagnostic tool or the CLI commands `ping mpls`, `ping mpls l2circuit`, `ping mpls l2vpn`, and `ping mpls l3vpn` to diagnose the state of label-switched paths (LSPs), Layer 2 and Layer 3 virtual private networks (VPNs), and Layer 2 circuits.

Based on how the LSP or VPN outbound (egress) node at the remote endpoint of the connection replies to the probes, you can determine the connectivity of the LSP or VPN.

Each probe is an echo request sent to the LSP or VPN exit point as an MPLS packet with a UDP payload. If the outbound node receives the echo request, it checks the contents of the probe and returns a value in the UDP payload of the response packet. If the device receives the response packet, it reports a successful ping response.

Responses that take longer than 2 seconds are identified as failed probes.

Table 166 on page 1456 summarizes the options for using either the J-Web ping MPLS diagnostic tool or the CLI `ping mpls` command to display information about MPLS connections in VPNs and LSPs.

Table 166: Options for Checking MPLS Connections

J-Web Ping MPLS Tool	ping mpls Command	Purpose	Additional Information
Ping RSVP-signaled LSP	`ping mpls rsvp`	Checks the operability of an LSP that has been set up by the Resource Reservation Protocol (RSVP). The device pings a particular LSP using the configured LSP name.	When an RSVP-signaled LSP has several paths, the device sends the ping requests on the path that is currently active.

Table 166: Options for Checking MPLS Connections *(Continued)*

J-Web Ping MPLS Tool	ping mpls Command	Purpose	Additional Information
Ping LDP-signaled LSP	`ping mpls ldp`	Checks the operability of an LSP that has been set up by the Label Distribution Protocol (LDP). The device pings a particular LSP using the forwarding equivalence class (FEC) prefix and length.	When an LDP-signaled LSP has several gateways, the device sends the ping requests through the first gateway. Ping requests sent to LDP-signaled LSPs use only the master routing instance.
Ping LSP to Layer 3 VPN prefix	`ping mpls l3vpn`	Checks the operability of the connections related to a Layer 3 VPN. The device tests whether a prefix is present in a provider edge (PE) device's VPN routing and forwarding (VRF) table, by means of a Layer 3 VPN destination prefix.	The device does not test the connection between a PE device and a customer edge (CE) router.
Locate LSP using interface name	`ping mpls l2vpn interface`	Checks the operability of the connections related to a Layer 2 VPN. The device directs outgoing request probes out the specified interface.	–
Instance to which this connection belongs	`ping mpls l2vpn instance`	Checks the operability of the connections related to a Layer 2 VPN. The device pings on a combination of the Layer 2 VPN routing instance name, the local site identifier, and the remote site identifier, to test the integrity of the Layer 2 VPN circuit (specified by the identifiers) between the inbound and outbound PE routers.	–

Table 166: Options for Checking MPLS Connections *(Continued)*

J-Web Ping MPLS Tool	ping mpls Command	Purpose	Additional Information
Locate LSP from interface name	`ping mpls l2circuit interface`	Checks the operability of the Layer 2 circuit connections. The device directs outgoing request probes out the specified interface.	–
Locate LSP from virtual circuit information	`ping mpls l2circuit virtual-circuit`	Checks the operability of the Layer 2 circuit connections. The device pings on a combination of the IPv4 prefix and the virtual circuit identifier on the outbound PE router, testing the integrity of the Layer 2 circuit between the inbound and outbound PE routers.	–
Ping end point of LSP	`ping mpls lsp-end-point`	Checks the operability of an LSP endpoint. The device pings an LSP endpoint using either an LDP FEC prefix or an RSVP LSP endpoint address.	–

Before using the ping MPLS feature, make sure that the receiving interface on the VPN or LSP remote endpoint has MPLS enabled, and that the loopback interface on the outbound node is configured as `127.0.0.1`. The source address for MPLS probes must be a valid address on the J Series device.

This section includes the following topics:

MPLS Enabled

To process ping MPLS requests, the remote endpoint of the VPN or LSP must be configured appropriately. You must enable MPLS on the receiving interface of the outbound node for the VPN or LSP. If MPLS is not enabled, the remote endpoint drops the incoming request packets and returns an “ICMP host unreachable” message to the J Series device.

Loopback Address

The loopback address (`lo0`) on the outbound node must be configured as `127.0.0.1`. If this interface address is not configured correctly, the outbound node does not have this forwarding entry. It drops the incoming request packets and returns a “host unreachable” message to the J Series device.

Source Address for Probes

The source IP address you specify for a set of probes must be an address configured on one of the J Series device interfaces. If it is not a valid J Series device address, the ping request fails with the error message “Can't assign requested address.”

Using the ping Command

You can perform certain tasks only through the CLI. Use the CLI `ping` command to verify that a host can be reached over the network. This command is useful for diagnosing host and network connectivity problems. The device sends a series of ICMP echo (ping) requests to a specified host and receives ICMP echo responses.

SEE ALSO

- ping
- ping mpls ldp
- ping mpls lsp-end-point
- ping mpls l2circuit
- ping mpls l2vpn
- ping mpls l3vpn
- ping mpls rsvp

Using Packet Capture to Analyze Network Traffic

IN THIS SECTION

Packet Capture Overview | 1460

- Example: Enable Packet Capture on a Device | **1463**
- Example: Configure Packet Capture on an Interface | **1468**
- Example: Configure a Firewall Filter for Packet Capture | **1471**
- Example: Configure Packet Capture for Datapath Debugging | **1474**
- Disable Packet Capture | **1479**
- Modify Encapsulation on Interfaces with Packet Capture Configured | **1479**
- Delete Packet Capture Files | **1481**
- Display Packet Headers | **1482**

Packet Capture Overview

IN THIS SECTION

- Packet Capture on Device Interfaces | **1461**
- Firewall Filters for Packet Capture | **1462**
- Packet Capture Files | **1462**
- Analysis of Packet Capture Files | **1463**

Packet capture is a tool that helps you to analyze network traffic and troubleshoot network problems. The packet capture tool captures real-time data packets traveling over the network for monitoring and logging.

NOTE: Packet capture is supported on physical interfaces, reth interfaces, and tunnel interfaces, such as gr, ip, st0, and lsq-/ls.

Packets are captured as binary data, without modification. You can read the packet information offline with a packet analyzer such as Wireshark or tcpdump. If you need to quickly capture packets destined for or originating from the Routing Engine and analyze them online, you can use the J-Web packet capture diagnostic tool.

NOTE: The packet capture tool does not support IPv6 packet capture.

You can use either the J-Web configuration editor or CLI configuration editor to configure packet capture.

Network administrators and security engineers use packet capture to perform the following tasks:

- Monitor network traffic and analyze traffic patterns.
- Identify and troubleshoot network problems.
- Detect security breaches in the network, such as unauthorized intrusions, spyware activity, or ping scans.

Packet capture operates like traffic sampling on the device, except that it captures entire packets including the Layer 2 header and saves the contents to a file in libpcap format. Packet capture also captures IP fragments.

You cannot enable packet capture and traffic sampling on the device at the same time. Unlike traffic sampling, there are no tracing operations for packet capture.

NOTE: You can enable packet capture and *port mirroring* simultaneously on a device.

This section contains the following topics:

Packet Capture on Device Interfaces

Packet capture is supported on the T1, T3, E1, E3, serial, Gigabit Ethernet, ADSL, G.SHDSL, PPPoE, and ISDN interfaces.

To capture packets on an ISDN interface, configure packet capture on the dialer interface. To capture packets on a PPPoE interface, configure packet capture on the PPPoE *logical interface*.

Packet capture supports PPP, Cisco HDLC, Frame Relay, and other ATM encapsulations. Packet capture also supports Multilink PPP (MLPPP), Multilink Frame Relay end-to-end (MLFR), and Multilink Frame Relay UNI/NNI (MFR) encapsulations.

You can capture all IPv4 packets flowing on an interface in the inbound or outbound direction. However, on traffic that bypasses the flow software module (protocol packets such as ARP, OSPF, and PIM), packets generated by the Routing Engine are not captured unless you have configured and applied a *firewall filter* on the interface in the outbound direction.

Tunnel interfaces support packet capture in the outbound direction only.

Use the J-Web configuration editor or CLI configuration editor to specify the maximum packet size, the filename to be used for storing the captured packets, the maximum file size, the maximum number of packet capture files, and the file permissions.

NOTE: For packets captured on T1, T3, E1, E3, serial, and ISDN interfaces in the outbound (egress) direction, the size of the packet captured might be 1 byte less than the maximum packet size configured because of the packet loss priority (PLP) bit.

To modify encapsulation on an interface with packet capture configured, you must disable packet capture.

Firewall Filters for Packet Capture

When you enable packet capture on a device, all packets flowing in the direction specified in packet capture configuration (inbound, outbound, or both) are captured and stored. Configuring an interface to capture all packets might degrade the performance of the device. You can control the number of packets captured on an interface with firewall filters and specify various criteria to capture packets for specific traffic flows.

You must also configure and apply appropriate firewall filters on the interface if you need to capture packets generated by the host device, because interface sampling does not capture packets originating from the host device.

Packet Capture Files

When packet capture is enabled on an interface, the entire packet including the Layer 2 header is captured and stored in a file. You can specify the maximum size of the packet to be captured, up to 1500 bytes. Packet capture creates one file for each physical interface.

File creation and storage take place in the following way. Suppose you name the packet capture file **pcap-file**. Packet capture creates multiple files (one per physical interface), suffixing each file with the name of the physical interface; for example, **pcap-file.fe-0.0.1** for the Gigabit Ethernet interface **fe-0.0.1**. When the file named **pcap-file.fe-0.0.1** reaches the maximum size, the file is renamed **pcap-file.fe-0.0.1.0**. When the file named **pcap-file.fe-0.0.1** reaches the maximum size again, the file named **pcap-file.fe-0.0.1.0** is renamed **pcap-file.fe-0.0.1.1** and **pcap-file.fe-0.0.1** is renamed **pcap-file.fe-0.0.1.0**. This process continues until the maximum number of files is exceeded and the oldest file is overwritten. The **pcap-file.fe-0.0.1** file is always the latest file.

Packet capture files are not removed even after you disable packet capture on an interface.

Analysis of Packet Capture Files

Packet capture files are stored in libpcap format in the `/var/tmp` directory. You can specify user or administrator privileges for the files.

Packet capture files can be opened and analyzed offline with tcpdump or any packet analyzer that recognizes the libpcap format. You can also use FTP or the Session Control Protocol (SCP) to transfer the packet capture files to an external device.

> **NOTE**: Disable packet capture before opening the file for analysis or transferring the file to an external device with FTP or SCP. Disabling packet capture ensures that the internal file buffer is flushed and all the captured packets are written to the file.

Example: Enable Packet Capture on a Device

IN THIS SECTION

- Requirements | **1463**
- Overview | **1463**
- Configuration | **1464**
- Verification | **1466**

This example shows how to enable packet capture on a device, to analyze network traffic and to troubleshoot network problems.

Requirements

Before you begin:

- Establish basic connectivity.
- Configure network interfaces. See Interfaces User Guide for Security Devices.

Overview

In this example, you set the maximum packet capture size in each file as 500 bytes. The range is from 68 through 1500, and the default is 68 bytes. You specify the target filename for the packet capture file as

pcap-file. You then specify the maximum number of files to capture as 100. The range is from 2 through 10,000, and the default is 10 files. You set the maximum size of each file to 1024 bytes. The range is from 1,024 through 104,857,600, and the default is 512,000 bytes. Finally, you specify that all users have permission to read the packet capture files.

Configuration

IN THIS SECTION

- Procedure | 1464

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set forwarding-options packet-capture maximum-capture-size 500
set forwarding-options packet-capture file filename pcap-file files 100 size 1024 world-readable
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To enable packet capture on a device:

1. Set the maximum packet capture size.

```
[edit]
user@host# edit forwarding-options
user@host# set packet-capture maximum-capture-size 500
```

2. Specify the target filename.

```
[edit forwarding-options]
user@host# set packet-capture file filename pcap-file
```

3. Specify the maximum number of files to capture.

```
[edit forwarding-options]
user@host# set packet-capture file files 100
```

4. Specify the maximum size of each file.

```
[edit forwarding-options]
user@host# set packet-capture file size 1024
```

5. Specify that all users have permission to read the file.

```
[edit forwarding-options]
user@host# set packet-capture file world-readable
```

Results

From configuration mode, confirm your configuration by entering the `run show forwarding-options` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# run show forwarding-options
packet-capture {
    file filename pcap-file files 100 size 1k world-readable;
    maximum-capture-size 500;
}
```

If you are done configuring the device, enter `commit` from configuration mode.

Verification

IN THIS SECTION

- Verifying the Packet Capture Configuration | 1466
- Verifying Captured Packets | 1466

Confirm that the configuration is working properly.

Verifying the Packet Capture Configuration

Purpose

Verify that the packet capture is configured on the device.

Action

From configuration mode, enter the `run show forwarding-options` command. Verify that the output shows the intended file configuration for capturing packets.

Verifying Captured Packets

Purpose

Verify that the packet capture file is stored under the `/var/tmp` directory and the packets can be analyzed offline.

Action

1. Disable packet capture.

   Using FTP, transfer a packet capture file (for example, `126b.fe-0.0.1`), to a server where you have installed packet analyzer tools (for example, `tools-server`).

   a. From configuration mode, connect to `tools-server` using FTP.

```
[edit]
user@host# run ftp tools-server
```

```
Connected to tools-server.mydomain.net
220 tools-server.mydomain.net FTP server (Version 6.00LS) ready
Name (tools-server:user):remoteuser
331 Password required for remoteuser.
Password:
230 User remoteuser logged in.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

b. Navigate to the directory where packet capture files are stored on the device.

```
ftp> lcd /var/tmp
Local directory now /cf/var/tmp
```

c. Copy the packet capture file that you want to analyze to the server, for example `126b.fe-0.0.1`.

```
ftp> put 126b.fe-0.0.1
local: 126b.fe-0.0.1 remote: 126b.fe-0.0.1
200 PORT command successful.
150 Opening BINARY mode data connection for '126b.fe-0.0.1'.
100%   1476       00:00 ETA
226 Transfer complete.
1476 bytes sent in 0.01 seconds (142.42 KB/s)
```

d. Return to configuration mode.

```
ftp> bye
221 Goodbye.
[edit]
user@host#
```

2. Open the packet capture file on the server with tcpdump or any packet analyzer that supports libpcap format and review the output.

```
root@server% tcpdump -r 126b.fe-0.0.1 -xevvvv
```

```
01:12:36.279769 Out 0:5:85:c4:e3:d1 > 0:5:85:c8:f6:d1, ethertype IPv4 (0x0800), length 98: (tos
0x0, ttl  64, id 33133, offset 0, flags [none], proto: ICMP (1), length: 84) 14.1.1.1 >
15.1.1.1: ICMP echo request seq 0, length 64
                         0005 85c8 f6d1 0005 85c4 e3d1 0800 4500
                         0054 816d 0000 4001 da38 0e01 0101 0f01
                         0101 0800 3c5a 981e 0000 8b5d 4543 51e6
                         0100 aaaa aaaa aaaa aaaa aaaa aaaa aaaa
                         aaaa aaaa 0000 0000 0000 0000 0000 0000
                         0000 0000 0000 0000 0000 0000 0000 0000
                         0000
01:12:36.279793 Out 0:5:85:c8:f6:d1 > 0:5:85:c4:e3:d1, ethertype IPv4 (0x0800), length 98: (tos
0x0, ttl  63, id 41227, offset 0, flags [none], proto: ICMP (1), length: 84) 15.1.1.1 >
14.1.1.1: ICMP echo reply seq 0, length 64
                         0005 85c4 e3d1 0005 85c8 f6d1 0800 4500
                         0054 a10b 0000 3f01 bb9a 0f01 0101 0e01
                         0101 0000 445a 981e 0000 8b5d 4543 51e6
                         0100 aaaa aaaa aaaa aaaa aaaa aaaa aaaa
                         aaaa aaaa 0000 0000 0000 0000 0000 0000
                         0000 0000 0000 0000 0000 0000 0000 0000
                         0000
root@server%
```

Example: Configure Packet Capture on an Interface

IN THIS SECTION

- Requirements | **1469**
- Overview | **1469**
- Configuration | **1469**
- Verification | **1470**

This example shows how to configure packet capture on an interface to analyze traffic.

Requirements

Before you begin:

- Establish basic connectivity.
- Configure network interfaces. See Interfaces User Guide for Security Devices.

Overview

In this example, you create an interface called fe-0/0/1 and then configure the direction of the traffic for which you are enabling packet capture on the logical interface as inbound and outbound.

NOTE: On traffic that bypasses the flow software module (protocol packets such as ARP, OSPF, and PIM), packets generated by the Routing Engine are not captured unless you have configured and applied a firewall filter on the interface in the output direction.

Configuration

IN THIS SECTION

- Procedure | 1469

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
edit interfaces fe-0/0/1
set unit 0 family inet sampling input output
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode*.

To configure packet capture on an interface:

1. Create an interface.

```
[edit]
user@host# edit interfaces fe-0/0/1
```

2. Configure the direction of the traffic.

```
[edit interfaces fe-0/0/1]
user@host# set unit 0 family inet sampling input output
```

3. If you are done configuring the device, commit the configuration.

```
[edit]
user@host# commit
```

Verification

IN THIS SECTION

- Verifying the Packet Capture Configuration | **1470**

Verifying the Packet Capture Configuration

Purpose

Confirm that the configuration is working properly.

Verify that packet capture is configured on the interface.

Action

From configuration mode, enter the `run show interfaces fe-0/0/1` command.

Example: Configure a Firewall Filter for Packet Capture

IN THIS SECTION

- Requirements | 1471
- Overview | 1471
- Configuration | 1472
- Verification | 1474

This example shows how to configure a firewall filter for packet capture and apply it to a logical interface.

Requirements

Before you begin:

- Establish basic connectivity.
- Configure network interfaces. See Interfaces User Guide for Security Devices.

Overview

IN THIS SECTION

- Topology | 1472

In this example, you set a firewall filter called dest-all and a term name called dest-term to capture packets from a specific destination address, which is 192.168.1.1/32. You define the match condition to accept the sampled packets. Finally, you apply the dest-all filter to all of the outgoing packets on interface fe-0/0/1.

NOTE: If you apply a firewall filter on the loopback interface, it affects all traffic to and from the Routing Engine. If the firewall filter has a `sample` action, packets to and from the Routing Engine are sampled. If packet capture is enabled, then packets to and from the Routing Engine are captured in the files created for the input and output interfaces.

Topology

Configuration

IN THIS SECTION

- Procedure | 1472

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set firewall filter dest-all term dest-term from destination-address 192.168.1.1/32
set firewall filter dest-all term dest-term then sample accept
edit interfaces
set interfaces fe-0/0/1 unit 0 family inet filter output dest-all
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide* .

To configure a firewall filter for packet capture and apply it to a logical interface:

1. Specify the firewall filter and its destination address.

```
[edit]
user@host# edit firewall
user@host# set filter dest-all term dest-term from destination-address 192.168.1.1/32
```

2. Define the match condition and its action.

```
[edit firewall]
user@host# set filter dest-all term dest-term then sample accept
```

3. Apply the filter to all the outgoing packets.

```
[edit interfaces]
user@host# set interfaces fe-0/0/1 unit 0 family inet filter output dest-all
```

Results

From configuration mode, confirm your configuration by entering the `run show firewall filter dest-all` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
[edit]
user@host# run show firewall filter dest-all
term dest-term {
    from {
        destination-address 192.168.1.1/32;
    }
    then {
        sample;
        accept;
    }
}
```

If you are done configuring the device, enter `commit` from configuration mode.

Verification

IN THIS SECTION

- Verifying the Firewall Filter for Packet Capture Configuration | 1474

Verifying the Firewall Filter for Packet Capture Configuration

Purpose

Confirm that the configuration is working properly.

Verify that the firewall filter for packet capture is configured.

Action

From configuration mode, enter the `run show firewall filter dest-all` command. Verify that the output shows the intended configuration of the firewall filter for capturing packets sent to the destination address.

Example: Configure Packet Capture for Datapath Debugging

IN THIS SECTION

- Requirements | 1475
- Overview | 1475
- Configuration | 1475
- Verification | 1478

This example shows how to configure packet capture to monitor traffic that passes through the device. Packet capture then dumps the packets into a PCAP file format that can be later examined by the tcpdump utility.

Requirements

Before you begin, see "Debugging the Data Path (CLI Procedure)" on page 1440.

Overview

A filter is defined to filter traffic; then an action profile is applied to the filtered traffic. The action profile specifies a variety of actions on the processing unit. One of the supported actions is packet dump, which sends the packet to the Routing Engine and stores it in proprietary form to be read using the `show security datapath-debug capture` command.

NOTE: Data path debugging is supported on SRX1400, SRX3400, SRX3600, SRX5400, SRX5600, and SRX5800.

Configuration

IN THIS SECTION

- Procedure | 1475

Procedure

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, copy and paste the commands into the CLI at the `[edit]` hierarchy level, and then enter `commit` from configuration mode.

```
set security datapath-debug capture-file my-capture
set security datapath-debug capture-file format pcap
set security datapath-debug capture-file size 1m
set security datapath-debug capture-file files 5
set security datapath-debug maximum-capture-size 400
set security datapath-debug action-profile do-capture event np-ingress packet-dump
set security datapath-debug packet-filter my-filter action-profile do-capture
set security datapath-debug packet-filter my-filter source-prefix 1.2.3.4/32
```

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For instructions on how to do that, see *Using the CLI Editor in Configuration Mode* in the *Junos OS CLI User Guide*.

To configure packet capture:

1. Edit the security datapath-debug option for the multiple processing units along the packet-processing path:

```
[edit]
user@host# edit security datapath-debug
```

2. Enable the capture file, the file format, the file size, and the number of files. Size number limits the size of the capture file. After the limit size is reached, if the file number is specified, then the capture file will be rotated to filename *x*, where *x* is auto-incremented until it reaches the specified index and then returns to zero. If no files index is specified, the packets will be discarded after the size limit is reached. The default size is 512 kilobytes.

```
[edit security datapath-debug]
user@host# set capture-file my-capture format pcap size 1m files 5
[edit security datapath-debug]
user@host# set maximum-capture-size 400
```

3. Enable action profile and set the event. Set the action profile as do-capture and the event type as np-ingress:

```
[edit security datapath-debug]
user@host# edit action-profile do-capture
[edit security datapath-debug action-profile do-capture]
user@host# edit event np-ingress
```

4. Enable packet dump for the action profile:

```
[edit security datapath-debug action-profile do-capture event np-ingress]
user@host# set packet-dump
```

5. Enable packet filter, action, and filter options. The packet filter is set to my-filter, the action profile is set to do-capture, and filter option is set to source-prefix 1.2.3.4/32.

```
[edit security datapath-debug]
user@host# set security datapath-debug packet-filter my-filter action-profile do-capture
```

```
[edit security datapath-debug]
user@host#  set security datapath-debug packet-filter my-filter source-prefix 1.2.3.4/32
```

Results

From configuration mode, confirm your configuration by entering the `show security datapath-debug` command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

```
security {
    datapath-debug {
        capture-file {
            my-capture
            format pcap
            size 1m
            files 5;
        }
    }
    maximum-capture-size 100;
    action-profile do-capture {
        event np-ingress {
             packet-dump
         }
      }
    packet-filter my-filter {
         source-prefix 1.2.3.4/32
        action-profile do-capture
     }
 }
```

If you are done configuring the device, enter `commit` from configuration mode.

Verification

IN THIS SECTION

- Verifying Packet Capture | 1478
- Verifying Data Path Debugging Capture | 1478
- Verifying Data Path Debugging Counter | 1479

Confirm that the configuration is working properly.

Verifying Packet Capture

Purpose

Verify if the packet capture is working.

Action

From operational mode, enter the `request security datapath-debug capture start` command to start packet capture and enter the `request security datapath-debug capture stop` command to stop packet capture.

To view the results, from CLI operational mode, access the local UNIX shell and navigate to the directory /var/log/my-capture. The result can be read by using the tcpdump utility.

Verifying Data Path Debugging Capture

Purpose

Verify the details of data path debugging capture file.

Action

From operational mode, enter the `show security datapath-debug capture` command.

```
user@host>show security datapath-debug capture
```

WARNING: When you are done troubleshooting, make sure to remove or deactivate all the traceoptions configurations (not limited to flow traceoptions) and the complete security datapath-debug configuration stanza. If any debugging configurations remain active, they will continue to use the device's CPU and memory resources.

Verifying Data Path Debugging Counter

Purpose

Verify the details of the data path debugging counter.

Action

From operational mode, enter the `show security datapath-debug counter` command.

Disable Packet Capture

You must disable packet capture before opening the packet capture file for analysis or transferring the file to an external device. Disabling packet capture ensures that the internal file buffer is flushed and all the captured packets are written to the file.

To disable packet capture, enter from configuration mode:

```
[edit forwarding-options]
user@host# set packet-capture disable
```

If you are done configuring the device, enter `commit` from configuration mode.

Modify Encapsulation on Interfaces with Packet Capture Configured

Before modifying the encapsulation on a device interface that is configured for packet capture, you must disable packet capture and rename the latest packet capture file. Otherwise, packet capture saves the packets with different encapsulations in the same packet capture file. Packet files containing packets with different encapsulations are not useful, because packet analyzer tools like tcpdump cannot analyze such files.

After modifying the encapsulation, you can safely reenable packet capture on the device.

To change the encapsulation on interfaces with packet capture configured:

1. Disable packet capture (see "Disabling Packet Capture" on page 1479).
2. Enter `commit` from configuration mode.
3. Rename the latest packet capture file on which you are changing the encapsulation with the `.chdsl` extension.
   a. From operational mode, access the local UNIX shell.

```
user@host> start shell
%
```

   b. Navigate to the directory where packet capture files are stored.

```
% cd /var/tmp
%
```

   c. Rename the latest packet capture file for the interface on which you are changing the encapsulation; for example `fe.0.0.0`.

```
% mv pcap-file.fe.0.0.0 pcap-file.fe.0.0.0.chdsl
%
```

   d. Return to operational mode.

```
% exit
user@host>
```

4. Change the encapsulation on the interface using the J-Web user interface or CLI configuration editor.
5. If you are done configuring the device, enter `commit` from configuration mode.
6. Reenable packet capture (see "Example: Enabling Packet Capture on a Device" on page 1463).
7. If you are done configuring the device, enter `commit` from configuration mode.

Delete Packet Capture Files

Deleting packet capture files from the /var/tmp directory only temporarily removes the packet capture files. Packet capture files for the interface are automatically created again the next time a packet capture configuration change is committed or as part of a packet capture file rotation.

To delete a packet capture file:

1. Disable packet capture (see "Disabling Packet Capture" on page 1479).
2. Delete the packet capture file for the interface.
   a. From operational mode, access the local UNIX shell.

```
user@host> start shell
%
```

   b. Navigate to the directory where packet capture files are stored.

```
% cd /var/tmp
%
```

   c. Delete the packet capture file for the interface; for example `pcap-file.fe.0.0.0`.

```
% rm pcap-file.fe.0.0.0
%
```

   d. Return to operational mode.

```
% exit
user@host>
```

3. Reenable packet capture (see "Example: Enabling Packet Capture on a Device" on page 1463).
4. If you are done configuring the device, enter `commit` from configuration mode.

Display Packet Headers

Enter the `monitor traffic` command to display packet headers transmitted through network interfaces with the following syntax:

> **NOTE**: Using the `monitor traffic` command can degrade system performance. We recommend that you use filtering options—such as `count` and `matching`—to minimize the impact to packet throughput on the system.

```
user@host> monitor traffic <absolute-sequence> <count number> <interface interface-name> <layer2-
headers> <matching "expression"> <no-domain-names> <no-promiscuous> <no-resolve> <no-timestamp>
<print-ascii> <print-hex> <size bytes> <brief | detail | extensive>
```

Table 167 on page 1482 describes the `monitor traffic` command options.

Table 167: CLI monitor traffic Command Options

Option	Description
`absolute-sequence`	(Optional) Displays the absolute TCP sequence numbers.
`count number`	(Optional) Displays the specified number of packet headers. Specify a value from `0` through `100,000`. The command quits and exits to the command prompt after this number is reached.
`interface interface-name`	(Optional) Displays packet headers for traffic on the specified interface. If an interface is not specified, the lowest numbered interface is monitored.
`layer2-headers`	(Optional) Displays the link-layer packet header on each line.
`matching "expression"`	(Optional) Displays packet headers that match an expression enclosed in quotation marks (" "). Table 168 on page 1484 through Table 170 on page 1487 list match conditions, logical operators, and arithmetic, binary, and relational operators you can use in the expression.

Table 167: CLI monitor traffic Command Options *(Continued)*

Option	Description
`no-domain-names`	(Optional) Suppresses the display of the domain name portion of the hostname.
`no-promiscuous`	(Optional) Specifies *not* to place the monitored interface in promiscuous mode. In promiscuous mode, the interface reads every packet that reaches it. In nonpromiscuous mode, the interface reads only the packets addressed to it.
`no-resolve`	(Optional) Suppresses the display of hostnames.
`no-timestamp`	(Optional) Suppresses the display of packet header timestamps.
`print-ascii`	(Optional) Displays each packet header in ASCII format.
`print-hex`	(Optional) Displays each packet header, except link-layer headers, in hexadecimal format.
`size` *bytes*	(Optional) Displays the number of bytes for each packet that you specify. If a packet header exceeds this size, the displayed packet header is truncated. The default value is `96`.
`brief`	(Optional) Displays minimum packet header information. This is the default.
`detail`	(Optional) Displays packet header information in moderate detail. For some protocols, you must also use the `size` option to see detailed information.
`extensive`	(Optional) Displays the most extensive level of packet header information. For some protocols, you must also use the `size` option to see extensive information.

To quit the `monitor traffic` command and return to the command prompt, press Ctrl-C.

To limit the packet header information displayed by the `monitor traffic` command, include the `matching` "*expression*" option. An expression consists of one or more match conditions listed in Table 168 on page

1484, enclosed in quotation marks (" "). You can combine match conditions by using the logical operators listed in Table 169 on page 1486 (shown in order of highest to lowest precedence).

For example, to display TCP or UDP packet headers, enter:

```
user@host> monitor traffic matching “tcp || udp”
```

To compare the following types of expressions, use the relational operators listed in Table 170 on page 1487 (listed from highest to lowest precedence):

- Arithmetic—Expressions that use the arithmetic operators listed in Table 170 on page 1487.
- Binary—Expressions that use the binary operators listed in Table 170 on page 1487.
- Packet data accessor—Expressions that use the following syntax:

  ```
  protocol [byte-offset <size>]
  ```

  Replace *protocol* with any protocol in Table 168 on page 1484. Replace *byte-offset* with the byte offset, from the beginning of the packet header, to use for the comparison. The optional *size* parameter represents the number of bytes examined in the packet header—1, 2, or 4 bytes.

  For example, the following command displays all multicast traffic:

  ```
  user@host> monitor traffic matching “ether[0] & 1 !=0”
  ```

Table 168: CLI monitor traffic Match Conditions

Match Condition	Description
Entity Type	
host [*address* | *hostname*]	Matches packet headers that contain the specified address or hostname. You can preprend any of the following protocol match conditions, followed by a space, to host: arp, ip, rarp, or any of the Directional match conditions.
network *address*	Matches packet headers with source or destination addresses containing the specified network address.

Table 168: CLI monitor traffic Match Conditions *(Continued)*

Match Condition	Description
`network` *address* `mask` *mask*	Matches packet headers containing the specified network address and subnet mask.
`port [`*port-number* `|` *port-name*`]`	Matches packet headers containing the specified source or destination TCP or UDP port number or port name.
Directional	
`destination`	Matches packet headers containing the specified destination. Directional match conditions can be prepended to any Entity Type match conditions, followed by a space.
`source`	Matches packet headers containing the specified source.
`source and destination`	Matches packet headers containing the specified source *and* destination.
`source or destination`	Matches packet headers containing the specified source *or* destination.
Packet Length	
`less` *bytes*	Matches packets with lengths less than or equal to the specified value, in bytes.
`greater` *bytes*	Matches packets with lengths greater than or equal to the specified value, in bytes.
Protocol	
`arp`	Matches all ARP packets.
`ether`	Matches all Ethernet frames.
`ether [broadcast	multicast]`	Matches broadcast or multicast Ethernet frames. This match condition can be prepended with `source` or `destination`.

Table 168: CLI monitor traffic Match Conditions *(Continued)*

Match Condition	Description
`ether protocol [`*address* `| (\arp | \ip | \rarp)`	Matches Ethernet frames with the specified address or protocol type. The arguments `arp`, `ip`, and `rarp` are also independent match conditions, so they must be preceded with a backslash (\) when used in the `ether protocol` match condition.
`icmp`	Matches all ICMP packets.
`ip`	Matches all IP packets.
`ip [broadcast | multicast]`	Matches broadcast or multicast IP packets.
`ip protocol [`*address* `| (\icmp | igrp | \tcp | \udp)]`	Matches IP packets with the specified address or protocol type. The arguments `icmp`, `tcp`, and `udp` are also independent match conditions, so they must be preceded with a backslash (\) when used in the `ip protocol` match condition.
`isis`	Matches all IS-IS routing messages.
`rarp`	Matches all RARP packets.
`tcp`	Matches all TCP packets.
`udp`	Matches all UDP packets.

Table 169: CLI monitor traffic Logical Operators

Logical Operator	Description
`!`	Logical NOT. If the first condition does not match, the next condition is evaluated.
`&&`	Logical AND. If the first condition matches, the next condition is evaluated. If the first condition does not match, the next condition is skipped.

Table 169: CLI monitor traffic Logical Operators *(Continued)*

Logical Operator	Description
||	Logical OR. If the first condition matches, the next condition is skipped. If the first condition does not match, the next condition is evaluated.
()	Group operators to override default precedence order. Parentheses are special characters, each of which must be preceded by a backslash (\).

Table 170: CLI monitor traffic Arithmetic, Binary, and Relational Operators

Operator	Description
Arithmetic Operator	
+	Addition operator.
-	Subtraction operator.
/	Division operator.
Binary Operator	
&	Bitwise AND.
*	Bitwise exclusive OR.
|	Bitwise inclusive OR.
Relational Operator	
<=	A match occurs if the first expression is less than or equal to the second.
>=	A match occurs if the first expression is greater than or equal to the second.

Table 170: CLI monitor traffic Arithmetic, Binary, and Relational Operators *(Continued)*

Operator	Description
<	A match occurs if the first expression is less than the second.
>	A match occurs if the first expression is greater than the second.
=	A match occurs if the first expression is equal to the second.
!=	A match occurs if the first expression is not equal to the second.

The following is sample output from the `monitor traffic` command:

```
user@host> monitor traffic count 4 matching “arp” detail
```

```
Listening on fe-0/0/0, capture size 96 bytes  15:04:16.276780  In arp who-has 193.1.1.1 tell
host1.site2.net 15:04:16.376848  In arp who-has host2.site2.net tell host1.site2.net
15:04:16.376887  In arp who-has 193.1.1.2 tell host1.site2.net 15:04:16.601923  In arp who-has
193.1.1.3 tell host1.site2.net
```

Troubleshooting Security Devices

IN THIS SECTION

- Troubleshooting DNS Name Resolution in Logical System Security Policies (Primary Administrators Only) | **1489**
- Troubleshooting the Link Services Interface | **1490**
- Troubleshooting Security Policies | **1503**

Troubleshooting DNS Name Resolution in Logical System Security Policies (Primary Administrators Only)

IN THIS SECTION

- Problem | 1489
- Cause | 1489
- Solution | 1489

Problem

Description

The address of a hostname in an address book entry that is used in a security policy might fail to resolve correctly.

Cause

Normally, address book entries that contain dynamic hostnames refresh automatically for SRX Series Firewalls. The TTL field associated with a DNS entry indicates the time after which the entry should be refreshed in the policy cache. Once the TTL value expires, the SRX Series Firewall automatically refreshes the DNS entry for an address book entry.

However, if the SRX Series Firewall is unable to obtain a response from the DNS server (for example, the DNS request or response packet is lost in the network or the DNS server cannot send a response), the address of a hostname in an address book entry might fail to resolve correctly. This can cause traffic to drop as no security policy or session match is found.

Solution

The primary administrator can use the `show security dns-cache` command to display DNS cache information on the SRX Series Firewall. If the DNS cache information needs to be refreshed, the primary administrator can use the `clear security dns-cache` command.

NOTE: These commands are only available to the primary administrator on devices that are configured for logical systems. This command is not available in user logical systems or on devices that are not configured for logical systems.

SEE ALSO

Understanding Logical Systems Security Policies

Troubleshooting the Link Services Interface

IN THIS SECTION

- Determine Which CoS Components Are Applied to the Constituent Links | **1490**
- Determine What Causes Jitter and Latency on the Multilink Bundle | **1493**
- Determine If LFI and Load Balancing Are Working Correctly | **1494**
- Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device | **1503**

To solve configuration problems on a link services interface:

Determine Which CoS Components Are Applied to the Constituent Links

IN THIS SECTION

- Problem | **1491**
- Solution | **1491**

Problem

Description

You are configuring a multilink bundle, but you also have traffic without MLPPP encapsulation passing through constituent links of the multilink bundle. Do you apply all CoS components to the constituent links, or is applying them to the multilink bundle enough?

Solution

You can apply a scheduler map to the multilink bundle and its constituent links. Although you can apply several CoS components with the scheduler map, configure only the ones that are required. We recommend that you keep the configuration on the constituent links simple to avoid unnecessary delay in transmission.

Table 1 shows the CoS components to be applied on a multilink bundle and its constituent links.

Table 171: CoS Components Applied on Multilink Bundles and Constituent Links

Cos Component	Multilink Bundle	Constituent Links	Explanation
Classifier	Yes	No	CoS classification takes place on the incoming side of the interface, not on the transmitting side, so no classifiers are needed on constituent links.
Forwarding class	Yes	No	Forwarding class is associated with a queue, and the queue is applied to the interface by a scheduler map. The queue assignment is predetermined on the constituent links. All packets from Q2 of the multilink bundle are assigned to Q2 of the constituent link, and packets from all the other queues are queued to Q0 of the constituent link.

Table 171: CoS Components Applied on Multilink Bundles and Constituent Links *(Continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Scheduler map	Yes	Yes	Apply scheduler maps on the multilink bundle and the constituent link as follows: • Transmit rate—Make sure that the relative order of the transmit rate configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. • Scheduler priority—Make sure that the relative order of the scheduler priority configured on Q0 and Q2 is the same on the constituent links as on the multilink bundle. • Buffer size—Because all non-LFI packets from the multilink bundle transit on Q0 of the constituent links, make sure that the buffer size on Q0 of the constituent links is large enough. • RED drop profile—Configure a RED drop profile on the multilink bundle only. Configuring the RED drop profile on the constituent links applies a back pressure mechanism that changes the buffer size and introduces variation. Because this behavior might cause fragment drops on the constituent links, make sure to leave the RED drop profile at the default settings on the constituent links.
Shaping rate for a per-unit scheduler or an interface-level scheduler	No	Yes	Because per-unit scheduling is applied only at the end point, apply this shaping rate to the constituent links only. Any configuration applied earlier is overwritten by the constituent link configuration.

Table 171: CoS Components Applied on Multilink Bundles and Constituent Links *(Continued)*

Cos Component	Multilink Bundle	Constituent Links	Explanation
Transmit-rate exact or queue-level shaping	Yes	No	The interface-level shaping applied on the constituent links overrides any shaping on the queue. Thus apply transmit-rate exact shaping on the multilink bundle only.
Rewrite rules	Yes	No	Rewrite bits are copied from the packet into the fragments automatically during fragmentation. Thus what you configure on the multilink bundle is carried on the fragments to the constituent links.
Virtual channel group	Yes	No	Virtual channel groups are identified through firewall filter rules that are applied on packets only before the multilink bundle. Thus you do not need to apply the virtual channel group configuration to the constituent links.

SEE ALSO

- Class of Service User Guide (Security Devices)

Determine What Causes Jitter and Latency on the Multilink Bundle

IN THIS SECTION

- Problem | **1494**
- Solution | **1494**

Problem

Description

To test jitter and latency, you send three streams of IP packets. All packets have the same IP precedence settings. After configuring LFI and CRTP, the latency increased even over a noncongested link. How can you reduce jitter and latency?

Solution

To reduce jitter and latency, do the following:

1. Make sure that you have configured a shaping rate on each constituent link.
2. Make sure that you have not configured a shaping rate on the link services interface.
3. Make sure that the configured shaping rate value is equal to the physical interface bandwidth.
4. If shaping rates are configured correctly, and jitter still persists, contact the Juniper Networks Technical Assistance Center (JTAC).

Determine If LFI and Load Balancing Are Working Correctly

IN THIS SECTION

- Problem | **1494**
- Solution | **1495**

Problem

Description

In this case, you have a single network that supports multiple services. The network transmits data and delay-sensitive voice traffic. After configuring MLPPP and LFI, make sure that voice packets are transmitted across the network with very little delay and jitter. How can you find out if voice packets are being treated as LFI packets and load balancing is performed correctly?

Solution

When LFI is enabled, data (non-LFI) packets are encapsulated with an MLPPP header and fragmented to packets of a specified size. The delay-sensitive, voice (LFI) packets are PPP-encapsulated and interleaved between data packet fragments. Queuing and load balancing are performed differently for LFI and non-LFI packets.

To verify that LFI is performed correctly, determine that packets are fragmented and encapsulated as configured. After you know whether a packet is treated as an LFI packet or a non-LFI packet, you can confirm whether the load balancing is performed correctly.

`Solution Scenario`—Suppose two Juniper Networks devices, R0 and R1, are connected by a multilink bundle `lsq-0/0/0.0` that aggregates two serial links, `se-1/0/0` and `se-1/0/1`. On R0 and R1, MLPPP and LFI are enabled on the link services interface and the fragmentation threshold is set to 128 bytes.

In this example, we used a packet generator to generate voice and data streams. You can use the packet capture feature to capture and analyze the packets on the incoming interface.

The following two data streams were sent on the multilink bundle:

- 100 data packets of 200 bytes (larger than the fragmentation threshold)
- 500 data packets of 60 bytes (smaller than the fragmentation threshold)

The following two voice streams were sent on the multilink bundle:

- 100 voice packets of 200 bytes from source port 100
- 300 voice packets of 200 bytes from source port 200

To confirm that LFI and load balancing are performed correctly:

> **NOTE**: Only the significant portions of command output are displayed and described in this example.

1. Verify packet fragmentation. From operational mode, enter the `show interfaces lsq-0/0/0` command to check that large packets are fragmented correctly.

```
user@R0#> show interfaces lsq-0/0/0
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
  Link-level type: LinkService, MTU: 1504
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps
```

```
  Last flapped   : 2006-08-01 10:45:13 PDT (2w0d 06:06 ago)
  Input rate     : 0 bps (0 pps)
  Output rate    : 0 bps (0 pps)

  Logical interface lsq-0/0/0.0 (Index 69) (SNMP ifIndex 42)
    Flags: Point-To-Point SNMP-Traps 0x4000 Encapsulation: Multilink-PPP
    Bandwidth: 16mbps
    Statistics         Frames      fps       Bytes        bps
    Bundle:
      Fragments:
        Input :             0        0           0          0
        Output:          1100        0      118800          0
      Packets:
        Input :             0        0           0          0
        Output:          1000        0      112000          0
   ...
    Protocol inet, MTU: 1500
      Flags: None
      Addresses, Flags: Is-Preferred Is-Primary
        Destination: 9.9.9/24, Local: 9.9.9.10
```

Meaning—The output shows a summary of packets transiting the device on the multilink bundle. Verify the following information on the multilink bundle:

- The total number of transiting packets = 1000
- The total number of transiting fragments=1100
- The number of data packets that were fragmented =100

The total number of packets sent (600 + 400) on the multilink bundle match the number of transiting packets (1000), indicating that no packets were dropped.

The number of transiting fragments exceeds the number of transiting packets by 100, indicating that 100 large data packets were correctly fragmented.

Corrective Action—If the packets are not fragmented correctly, check your fragmentation threshold configuration. Packets smaller than the specified fragmentation threshold are not fragmented.

2. Verify packet encapsulation. To find out whether a packet is treated as an LFI or non-LFI packet, determine its encapsulation type. LFI packets are PPP encapsulated, and non-LFI packets are encapsulated with both PPP and MLPPP. PPP and MLPPP encapsulations have different overheads resulting in different-sized packets. You can compare packet sizes to determine the encapsulation type.

A small unfragmented data packet contains a PPP header and a single MLPPP header. In a large fragmented data packet, the first fragment contains a PPP header and an MLPPP header, but the consecutive fragments contain only an MLPPP header.

PPP and MLPPP encapsulations add the following number of bytes to a packet:

- PPP encapsulation adds 7 bytes:

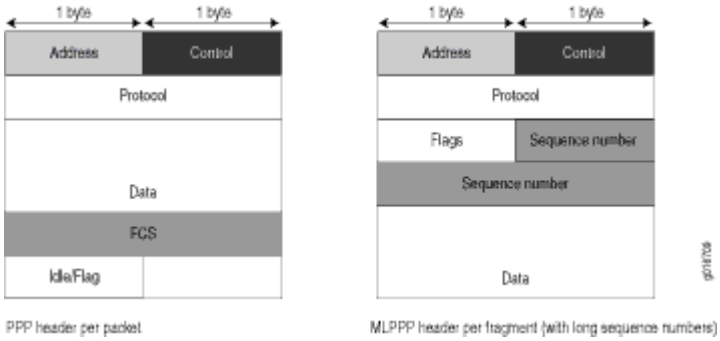
  4 bytes of header+2 bytes of frame check sequence (FCS)+1 byte that is idle or contains a flag

- MLPPP encapsulation adds between 6 and 8 bytes:

  4 bytes of PPP header+2 to 4 bytes of multilink header

Figure 1 shows the overhead added to PPP and MLPPP headers.

Figure 48: PPP and MLPPP Headers



For CRTP packets, the encapsulation overhead and packet size are even smaller than for an LFI packet. For more information, see Example: Configuring the Compressed Real-Time Transport Protocol.

Table 2 shows the encapsulation overhead for a data packet and a voice packet of 70 bytes each. After encapsulation, the size of the data packet is larger than the size of the voice packet.

Table 172: PPP and MLPPP Encapsulation Overhead

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Voice packet (LFI)	PPP	70 bytes	4 + 2 + 1 = 7 bytes	77 bytes

Table 172: PPP and MLPPP Encapsulation Overhead *(Continued)*

Packet Type	Encapsulation	Initial Packet Size	Encapsulation Overhead	Packet Size after Encapsulation
Data fragment (non-LFI) with short sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 2 = 13 bytes	83 bytes
Data fragment (non-LFI) with long sequence	MLPPP	70 bytes	4 + 2 + 1 + 4 + 4 = 15 bytes	85 bytes

From operational mode, enter the `show interfaces queue` command to display the size of transmitted packet on each queue. Divide the number of bytes transmitted by the number of packets to obtain the size of the packets and determine the encapsulation type.

3. Verify load balancing. From operational mode, enter the `show interfaces queue` command on the multilink bundle and its constituent links to confirm whether load balancing is performed accordingly on the packets.

```
user@R0> show interfaces queue lsq-0/0/0
Physical interface: lsq-0/0/0, Enabled, Physical link is Up
  Interface index: 136, SNMP ifIndex: 29
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :               600                 0 pps
    Bytes                :             44800                 0 bps
  Transmitted:
    Packets              :               600                 0 pps
    Bytes                :             44800                 0 bps
    Tail-dropped packets :                 0                 0 pps
    RED-dropped packets  :                 0                 0 pps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                 0                 0 pps
    Bytes                :                 0                 0 bps
```

```
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                400                0 pps
    Bytes                :              61344                0 bps
  Transmitted:
    Packets              :                400                0 pps
    Bytes                :              61344                0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                  0                0 pps
    Bytes                :                  0                0 bps
 …
```

```
user@R0> show interfaces queue se-1/0/0
Physical interface: se-1/0/0, Enabled, Physical link is Up
  Interface index: 141, SNMP ifIndex: 35
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :                350                0 pps
    Bytes                :              24350                0 bps
  Transmitted:
    Packets              :                350                0 pps
    Bytes                :              24350                0 bps
   ...
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                  0                0 pps
    Bytes                :                  0                0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :                100                0 pps
    Bytes                :              15272                0 bps
  Transmitted:
    Packets              :                100                0 pps
    Bytes                :              15272                0 bps
    …
```

```
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                19                0 pps
    Bytes                :               247                0 bps
  Transmitted:
    Packets              :                19                0 pps
    Bytes                :               247                0 bps
    …
```

```
user@R0> show interfaces queue se-1/0/1
Physical interface: se-1/0/1, Enabled, Physical link is Up
  Interface index: 142, SNMP ifIndex: 38
Forwarding classes: 8 supported, 8 in use
Egress queues: 8 supported, 8 in use
Queue: 0, Forwarding classes: DATA
  Queued:
    Packets              :               350                0 pps
    Bytes                :             24350                0 bps
  Transmitted:
    Packets              :               350                0 pps
    Bytes                :             24350                0 bps
    …
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    Packets              :                 0                0 pps
    Bytes                :                 0                0 bps
  …
Queue: 2, Forwarding classes: VOICE
  Queued:
    Packets              :               300                0 pps
    Bytes                :             45672                0 bps
  Transmitted:
    Packets              :               300                0 pps
    Bytes                :             45672                0 bps
    …
Queue: 3, Forwarding classes: NC
  Queued:
    Packets              :                18                0 pps
    Bytes                :               234                0 bps
  Transmitted:
```

```
    Packets              :              18              0 pps
    Bytes                :             234              0 bps
```

Meaning—The output from these commands shows the packets transmitted and queued on each queue of the link services interface and its constituent links. Table 3 shows a summary of these values. (Because the number of transmitted packets equaled the number of queued packets on all the links, this table shows only the queued packets.)

Table 173: Number of Packets Transmitted on a Queue

Packets Queued	Bundle lsq-0/0/0.0	Constituent Link se-1/0/0	Constituent Link se-1/0/1	Explanation
Packets on Q0	600	350	350	The total number of packets transiting the constituent links (350+350 = 700) exceeded the number of packets queued (600) on the multilink bundle.
Packets on Q2	400	100	300	The total number of packets transiting the constituent links equaled the number of packets on the bundle.
Packets on Q3	0	19	18	The packets transiting Q3 of the constituent links are for keepalive messages exchanged between constituent links. Thus no packets were counted on Q3 of the bundle.

On the multilink bundle, verify the following:

- The number of packets queued matches the number transmitted. If the numbers match, no packets were dropped. If more packets were queued than were transmitted, packets were dropped because the buffer was too small. The buffer size on the constituent links controls congestion at the output stage. To correct this problem, increase the buffer size on the constituent links.
- The number of packets transiting Q0 (600) matches the number of large and small data packets received (100+500) on the multilink bundle. If the numbers match, all data packets correctly transited Q0.

- The number of packets transiting Q2 on the multilink bundle (400) matches the number of voice packets received on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

On the constituent links, verify the following:

- The total number of packets transiting Q0 (350+350) matches the number of data packets and data fragments (500+200). If the numbers match, all the data packets after fragmentation correctly transited Q0 of the constituent links.

  Packets transited both constituent links, indicating that load balancing was correctly performed on non-LFI packets.

- The total number of packets transiting Q2 (300+100) on constituent links matches the number of voice packets received (400) on the multilink bundle. If the numbers match, all voice LFI packets correctly transited Q2.

  LFI packets from source port `100` transited `se-1/0/0`, and LFI packets from source port `200` transited `se-1/0/1`. Thus all LFI (Q2) packets were hashed based on the source port and correctly transited both constituent links.

`Corrective Action`—If the packets transited only one link, take the following steps to resolve the problem:

a. Determine whether the physical link is `up` (operational) or `down` (unavailable). An unavailable link indicates a problem with the PIM, interface port, or physical connection (link-layer errors). If the link is operational, move to the next step.

b. Verify that the classifiers are correctly defined for non-LFI packets. Make sure that non-LFI packets are not configured to be queued to Q2. All packets queued to Q2 are treated as LFI packets.

c. Verify that at least one of the following values is different in the LFI packets: source address, destination address, IP protocol, source port, or destination port. If the same values are configured for all LFI packets, the packets are all hashed to the same flow and transit the same link.

4. Use the results to verify load balancing.

Determine Why Packets Are Dropped on a PVC Between a Juniper Networks Device and a Third-Party Device

IN THIS SECTION

- Problem | 1503
- Solution | 1503

Problem

Description

You are configuring a permanent virtual circuit (PVC) between T1, E1, T3, or E3 interfaces on a Juniper Networks device and a third-party device, and packets are being dropped and ping fails.

Solution

If the third-party device does not have the same FRF.12 support as the Juniper Networks device or supports FRF.12 in a different way, the Juniper Networks device interface on the PVC might discard a fragmented packet containing FRF.12 headers and count it as a "Policed Discard."

As a workaround, configure multilink bundles on both peers, and configure fragmentation thresholds on the multilink bundles.

Troubleshooting Security Policies

IN THIS SECTION

- Synchronizing Policies Between Routing Engine and Packet Forwarding Engine | 1504
- Checking a Security Policy Commit Failure | 1505
- Verifying a Security Policy Commit | 1505
- Debugging Policy Lookup | 1506

Synchronizing Policies Between Routing Engine and Packet Forwarding Engine

IN THIS SECTION

- Problem | 1504
- Solution | 1504

Problem

Description

Security policies are stored in the routing engine and the packet forwarding engine. Security policies are pushed from the Routing Engine to the Packet Forwarding Engine when you commit configurations. If the security policies on the Routing Engine are out of sync with the Packet Forwarding Engine, the commit of a configuration fails. Core dump files may be generated if the commit is tried repeatedly. The out of sync can be due to:

- A policy message from Routing Engine to the Packet Forwarding Engine is lost in transit.
- An error with the routing engine, such as a reused policy UID.

Environment

The policies in the Routing Engine and Packet Forwarding Engine must be in sync for the configuration to be committed. However, under certain circumstances, policies in the Routing Engine and the Packet Forwarding Engine might be out of sync, which causes the commit to fail.

Symptoms

When the policy configurations are modified and the policies are out of sync, the following error message displays - `error: Warning: policy might be out of sync between RE and PFE <SPU-name(s)> Please request security policies check/resync.`

Solution

Use the `show security policies checksum` command to display the security policy checksum value and use the `request security policies resync` command to synchronize the configuration of security policies in the Routing Engine and Packet Forwarding Engine, if the security policies are out of sync.

SEE ALSO

show security policies checksum

request security policies check

request security policies resync

Checking a Security Policy Commit Failure

IN THIS SECTION

- Problem | 1505
- Solution | 1505

Problem

Description

Most policy configuration failures occur during a commit or runtime.

Commit failures are reported directly on the CLI when you execute the CLI command **commit-check** in configuration mode. These errors are configuration errors, and you cannot commit the configuration without fixing these errors.

Solution

To fix these errors, do the following:

1. Review your configuration data.
2. Open the file /var/log/nsd_chk_only. This file is overwritten each time you perform a commit check and contains detailed failure information.

Verifying a Security Policy Commit

IN THIS SECTION

- Problem | 1506
- Solution | 1506

Problem

Description

Upon performing a policy configuration commit, if you notice that the system behavior is incorrect, use the following steps to troubleshoot this problem:

Solution

1. Operational **show** Commands—Execute the operational commands for security policies and verify that the information shown in the output is consistent with what you expected. If not, the configuration needs to be changed appropriately.
2. Traceoptions—Set the `traceoptions` command in your policy configuration. The flags under this hierarchy can be selected as per user analysis of the `show` command output. If you cannot determine what flag to use, the flag option `all` can be used to capture all trace logs.

```
user@host# set security policies traceoptions <flag all>
```

You can also configure an optional filename to capture the logs.

```
user@host# set security policies traceoptions <filename>
```

If you specified a filename in the trace options, you can look in the /var/log/<*filename*> for the log file to ascertain if any errors were reported in the file. (If you did not specify a filename, the default filename is eventd.) The error messages indicate the place of failure and the appropriate reason.

After configuring the trace options, you must recommit the configuration change that caused the incorrect system behavior.

Debugging Policy Lookup

IN THIS SECTION

- Problem | **1507**
- Solution | **1507**

Problem

Description

When you have the correct configuration, but some traffic was incorrectly dropped or permitted, you can enable the `lookup` flag in the security policies traceoptions. The `lookup` flag logs the lookup related traces in the trace file.

Solution

```
user@host# set security policies traceoptions <flag lookup>
```

PART 12

Configuration Statements and Operational Commands

Junos CLI Reference Overview | 1509

Junos CLI Reference Overview

We've consolidated all Junos CLI commands and configuration statements in one place. Learn about the syntax and options that make up the statements and commands and understand the contexts in which you'll use these CLI elements in your network configurations and operations.

- *Junos CLI Reference*

Click the links to access Junos OS and Junos OS Evolved configuration statement and command summary topics.

- *Configuration Statements*
- *CLI Commands*