JUNIPER NETWORKS | Engineering Simplicity

Junos® OS

Monitoring, Sampling, and Collection Services Interfaces User Guide

Published
2020-03-31

JUNOS

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

Junos® OS Monitoring, Sampling, and Collection Services Interfaces User Guide

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 the Documentation

IN THIS SECTION

Use this guide to configure traffic flow monitoring, packet flow capture, traffic sampling for accounting or discard, port mirroring to an external device, and real-time performance monitoring.

Documentation and Release Notes

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

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

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

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

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

Table 1: Notice Icons

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

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

Table 2: Text and Syntax Conventions

Convention	Description	Examples
Bold text like this	Represents text that you type.	To enter configuration mode, type the **configure** command: user@host> **configure**
`Fixed-width text like this`	Represents output that appears on the terminal screen.	`user@host>` **show chassis alarms** `No alarms currently active`
Italic text like this	• Introduces or emphasizes important new terms. • Identifies guide names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS CLI User Guide* • RFC 1997, *BGP Communities Attribute*

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

Convention	Description	Examples
Italic text like this	Represents variables (options for which you substitute a value) in commands or configuration statements.	Configure the machine's domain name: [edit] root@# **set system domain-name** *domain-name*
Text like this	Represents names of configuration statements, commands, files, and directories; configuration hierarchy levels; or labels on routing platform components.	• To configure a stub area, include the **stub** statement at the **[edit protocols ospf area area-id]** hierarchy level. • The console port is labeled **CONSOLE**.
< > (angle brackets)	Encloses optional keywords or variables.	**stub <default-metric *metric*>;**
| (pipe symbol)	Indicates a choice between the mutually exclusive keywords or variables on either side of the symbol. The set of choices is often enclosed in parentheses for clarity.	**broadcast | multicast** **(*string1* | *string2* | *string3*)**
# (pound sign)	Indicates a comment specified on the same line as the configuration statement to which it applies.	**rsvp { # Required for dynamic MPLS only**
[ ] (square brackets)	Encloses a variable for which you can substitute one or more values.	**community name members [*community-ids*]**

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

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

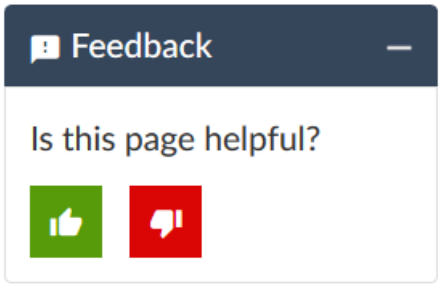
Table 2: Text and Syntax Conventions *(continued)*

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

Documentation Feedback

We encourage you to provide feedback so that we can improve our documentation. You can use either of the following methods:

- Online feedback system—Click TechLibrary Feedback, on the lower right of any page on the Juniper Networks TechLibrary site, and do one of the following:



  - Click the thumbs-up icon if the information on the page was helpful to you.
  - Click the thumbs-down icon if the information on the page was not helpful to you or if you have suggestions for improvement, and use the pop-up form to provide feedback.
- E-mail—Send your comments to techpubs-comments@juniper.net. Include the document or topic name, URL or page number, and software version (if applicable).

Requesting Technical Support

Technical product support is available through the Juniper Networks Technical Assistance Center (JTAC). If you are a customer with an active Juniper Care or Partner Support Services support contract, or are

covered under warranty, and need post-sales technical support, you can access our tools and resources online or open a case with JTAC.

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: https://www.juniper.net/customers/support/
- Search for known bugs: https://prsearch.juniper.net/
- Find product documentation: https://www.juniper.net/documentation/
- Find solutions and answer questions using our Knowledge Base: https://kb.juniper.net/
- Download the latest versions of software and review release notes: https://www.juniper.net/customers/csc/software/
- Search technical bulletins for relevant hardware and software notifications: https://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: https://www.juniper.net/company/communities/
- Create a service request online: https://myjuniper.juniper.net

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

Creating a Service Request with JTAC

You can create a service request with JTAC on the Web or by telephone.

- Visit https://myjuniper.juniper.net.
- Call 1-888-314-JTAC (1-888-314-5822 toll-free in the USA, Canada, and Mexico).

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

PART 1

Flow Monitoring and Flow Collection Services

CHAPTER 1

Understanding Flow Monitoring

IN THIS CHAPTER

Flow Monitoring Terms and Acronyms

A

active flow monitoring — Technique to lawfully intercept and observe specified data network traffic on an active router participating in the network.

Adaptive Services PIC — Advanced PIC that handles active flow monitoring, Network Address Translation (NAT), stateful firewall, and intrusion detection functions. For more information on the Adaptive Services PIC, see the *Junos Services Interfaces Configuration Guide*.

C

cflowd — Version 5 and version 8 flow monitoring process that captures flow information from network traffic and exports this data into summary tables. Once captured, flow data can be analyzed as needed. For more information about cflowd, see **http://www.caida.org**.

content destination — A recipient of monitored packets sent by a DTCP or dynamic flow capture-enabled monitoring station.

control source — A dynamic flow capture client that wants to monitor electronic data or voice transfer over the network. The control source sends filter requests to the dynamic flow capture-enabled monitoring station by using DTCP.

D

DTCP (Dynamic Tasking Control Protocol) — Protocol used to specify filtering criteria in a dynamic flow capture environment.

dynamic flow capture — Technique that allows DTCP-enabled control sources to send specified filtering criteria in real time to a monitoring station. The monitoring station passively monitors the specified traffic flows on demand and sends the captured packets to content destinations.

E

ES PIC — PIC that handles encryption and security services (such as IP Security [IPSec]).

F

flow collector interface — Converted Monitoring Services II PIC that processes multiple flow records into compressed ASCII data files and exports these files to an FTP server.

M

Monitoring Services II PIC — Advanced PIC that handles passive flow monitoring functions.

Monitoring Services III PIC — Advanced PIC that handles dynamic flow capture functions.

Monitoring Services PIC — Original PIC that handles passive and active flow monitoring functions.

MultiServices 100 PIC — Also referred to as MultiServices PIC Type 1. Advanced PIC that handles active flow capture functions.

MultiServices 400 PIC — Also referred to as MultiServices PIC Type 2. Advanced PIC that handles active flow capture functions.

MultiServices 500 PIC — Also referred to as MultiServices PIC Type 3. Advanced PIC that handles active flow capture functions.

P

passive flow monitoring — Technique to lawfully intercept and observe specified data network traffic on a passive flow monitoring station not participating in the network.

Configuring Flow Monitoring

IN THIS SECTION

The flow-monitoring application performs traffic flow monitoring and enables lawful interception of traffic between two routers or switches. Traffic flows can either be passively monitored by an offline router or switch or actively monitored by a router participating in the network.

To configure flow monitoring you need to do the following:

Configuring Flow-Monitoring Interfaces

To enable flow monitoring on the Monitoring Services PIC, include the **mo-*fpc*/*pic*/*port*** statement at the **[edit interfaces]** hierarchy level:

```
mo-fpc/pic/port {
    unit logical-unit-number {
        family inet {
            address address {
                destination address;
            }
            filter {
                group filter-group-number;
                input filter-name;
                output filter-name;
            }
            sampling {
                [ input output ];
            }
        }
    }
    multiservice-options {
        (core-dump | no-core-dump);
        (syslog | no-syslog);
        flow-control-options {
            down-on-flow-control;
            dump-on-flow-control;
            reset-on-flow-control;
        }
    }
}
```

Specify the physical and logical location of the flow-monitoring interface. You cannot use **unit 0**, because it is already used by internal processes. Specify the source and destination addresses. The **filter** statement allows you to associate an input or output filter or a filter group that you have already configured for this purpose. The **sampling** statement specifies the traffic direction: **input**, **output**, or both.

The **multiservice-options** statement allows you to configure properties related to flow-monitoring interfaces:

- Include the **core-dump** statement to enable storage of core files in **/var/tmp**.
- Include the **syslog** statement to enable storage of system logging information in **/var/log**.

> **NOTE:** Boot images for monitoring services interfaces are specified at the **[edit chassis images pic]** hierarchy level. You must include the following configuration to make the flow monitoring feature operable:
>
> ```
> [edit system]
> ntp {
>    boot-server ntp.example.net;
>    server 172.17.28.5;
> }
> processes {
>    ntp enable;
> }
> ```
>
> For more information, see the *Junos OS Administration Library*.

- Include the **flow-control-options** statement to configure flow control.

> **NOTE:** Starting with Junos OS Release 15.1, the multiservices PIC management daemon core file is generated when a prolonged flow control failure occurs and when you configure the setting to generate a core dump during prolonged flow control (by using the **dump-on-flow-control** option with the **flow-control-options** statement). The watchdog functionality continues to generate a kernel core file in such scenarios. In Junos OS Release 14.2 and earlier, an eJunos kernel core file is generated when a prolonged flow control failure occurs and when you configure the setting to generate a core dump during prolonged flow control.

Configuring Flow-Monitoring Properties

IN THIS SECTION

To configure flow-monitoring properties, include the **monitoring** statement at the **[edit forwarding-options]** hierarchy level:

```
monitoring name {
    family inet {
        output {
            cflowd hostname port port-number;
            export-format format;
            flow-active-timeout seconds;
            flow-export-destination {
                collector-pic;
            }
            flow-inactive-timeout seconds;
            interface interface-name {
                engine-id number;
                engine-type number;
                input-interface-index number;
                output-interface-index number;
                source-address address;
            }
        }
    }
```

A monitoring instance is a named entity that specifies collector information under the **monitoring *name*** statement. The following sections describe the properties you can configure:

Directing Traffic to Flow-Monitoring Interfaces

To direct traffic to a flow-monitoring interface, include the **interface** statement at the **[edit forwarding-options monitoring *name* output]** hierarchy level. By default, the Junos OS automatically assigns values for the **engine-id** and **engine-type** statements:

- **engine-id**—Monitoring interface location.
- **engine-type**—Platform-specific monitoring interface type.

The **source-address** statement specifies the traffic source for transmission of cflowd information; you must configure it manually. If you provide a different **source-address** statement for each monitoring services output interface, you can track which interface processes a particular cflowd record.

By default, the **input-interface-index** value is the SNMP index of the input interface. You can override the default by including a specific value. The **input-interface-index** and **output-interface-index** values are exported in fields present in the cflowd version 5 flow format.

Exporting Flows

To direct traffic to a flow collection interface, include the **flow-export-destination** statement. For more information about flow collection, see "Active Flow Monitoring Overview" on page 45.

To configure the cflowd version number, include the **export-format** statement at the **[edit forwarding-options monitoring *name* output]** hierarchy level. By default, version 5 is used. Version 8 enables the router software to aggregate the flow information using broader criteria and reduce cflowd traffic. Version 8 aggregation is performed periodically (every few seconds) on active flows and when flows are allowed to expire. Because the aggregation is performed periodically, active timeout events are ignored.

For more information on cflowd properties, see "Enabling Flow Aggregation" on page 489.

Configuring Time Periods When Flow Monitoring Is Active and Inactive

To configure time periods for active flow monitoring and intervals of inactivity, include the **flow-active-timeout** and **flow-inactive-timeout** statements at the **[edit forwarding-options monitoring *name* output]** hierarchy level:

- The **flow-active-timeout** statement specifies the time interval between flow exports for active flows. If the interval between the time the last packet was received and the time the flow was last exported exceeds the configured value, the flow is exported.

  This timer is needed to provide periodic updates when a flow has a long duration. The active timeout setting enables the router to retain the start time for the flow as a constant and send out periodic cflowd reports. This in turn allows the collector to register the start time and determine that a flow has survived for a duration longer than the configured active timeout.

  > **NOTE:** In active flow monitoring, the cflowd records are exported after a time period that is a multiple of 60 seconds and greater than or equal to the configured active timeout value. For example, if the active timeout value is 90 seconds, the cflowd records are exported at 120-second intervals. If the active timeout value is 150 seconds, the cflowd records are exported at 180-second intervals, and so forth.

- The **flow-inactive-timeout** statement specifies the interval of inactivity for a flow that triggers the flow export. If the interval between the current time and the time that the last packet for this flow was received exceeds the configured inactive timeout value, the flow is allowed to expire.

  If the flow stops transmitting for longer than the configured inactive timeout value, the router or switch purges it from the flow table and exports the cflowd record. As a result, the flow is forgotten as far as the PIC is concerned and if the same 5-tuple appears again, it is assigned a new start time and considered a new flow.

Both timers are necessary. The active timeout setting is needed to provide information for flows that constantly transmit packets for a long duration. The inactive timeout setting enables the router or switch to purge flows that have become inactive and that can waste tracking resources.

NOTE: The router must contain an Adaptive Services, Multiservices, or Monitoring Services PIC for the **flow-active-timeout** and **flow-inactive-timeout** statements to take effect.

Example: Configuring Flow Monitoring

The following is an example of flow-monitoring properties configured to support input SONET/SDH interfaces, output monitoring services interfaces, and export to cflowd for flow analysis. To complete the configuration, you also need to configure the interfaces and set up a virtual private network (VPN) routing and forwarding (VRF) instance. For information on cflowd, see "Enabling Flow Aggregation" on page 489.

```
[edit forwarding-options]
monitoring group1 {
   family inet {
      output {
         cflowd 192.168.245.2 port 2055;
         export-format cflowd-version-5;
         flow-active-timeout 60;
         flow-inactive-timeout 30;
         interface mo-4/0/0.1 {
            engine-id 1;
            engine-type 1;
            input-interface-index 44;
            output-interface-index 54;
            source-address 192.168.245.1;
         }
         interface mo-4/1/0.1 {
            engine-id 2;
            engine-type 1;
            input-interface-index 45;
            output-interface-index 55;
            source-address 192.168.245.1;
         }
         interface mo-4/2/0.1 {
            engine-id 3;
            engine-type 1;
            input-interface-index 46;
            output-interface-index 56;
            source-address 192.168.245.1;
         }
         interface mo-4/3/0.1 {
            engine-id 4;
```

```
            engine-type 1;
            input-interface-index 47;
            output-interface-index 57;
            source-address 192.168.245.1;
        }
    }
  }
}
```

Release History Table

Release	Description
15.1	Starting with Junos OS Release 15.1, the multiservices PIC management daemon core file is generated when a prolonged flow control failure occurs and when you configure the setting to generate a core dump during prolonged flow control (by using the **dump-on-flow-control** option with the **flow-control-options** statement).

RELATED DOCUMENTATION

- Active Flow Monitoring Overview | 45
- Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | 540
- Configuring Services Interface Redundancy with Flow Monitoring | 61
- Example: Configuring Active Monitoring on an M, MX or T Series Router's Logical System | 48

Flow Monitoring Output Formats

When you implement passive flow monitoring and active flow monitoring, you should be familiar with flow monitoring formats and fields. Version 5 and version 8 export data into specified fields. Version 9 exports data into templates.

The flow monitoring station monitors the traffic flow and exports the data in flow format to an external server. The Junos OS collects information about the following fields:

- Source and destination IP address
- Total number of bytes and packets sent
- Start and end times of the data flow
- Source and destination port numbers

- TCP flags
- IP protocol and IP type of service
- Originating AS of source and destination address
- Source and destination address prefix mask lengths
- Next-hop router's IP address
- MPLS label (version 9 only)
- ICMP (version 9 only)

Detailed descriptions of the formats are available as follows:

- Flow Monitoring Version 5 Format Output Fields on page 10
- Flow Monitoring Version 8 Format Output Fields on page 14
- Flow Monitoring Version 9 Format Output Fields on page 21

Flow Monitoring Version 5 Format Output Fields

A detailed explanation of version 5 packet formats and fields is shown in the following figures and tables:

- Figure 1 on page 10
- Table 3 on page 11
- Figure 2 on page 12
- Table 4 on page 12

Figure 1: Version 5 Packet Header Format

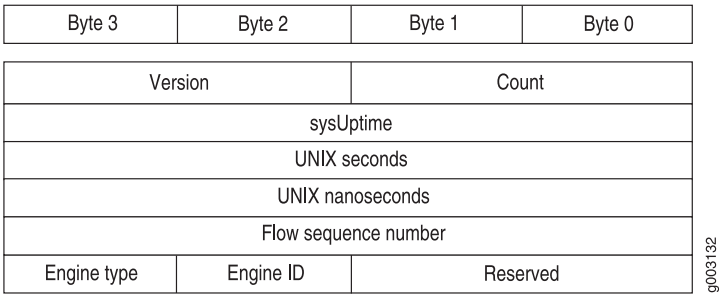


Table 3: Export Version 5 Packet Header Fields

Field	Description	Comments
Version	5	–
Count	The number of records in the Protocol Data Unit (PDU) or packet	–
sysUptime	Current time elapsed, in milliseconds, since the router started	–
UNIX seconds	Current seconds since 0000 UTC 1970	NTP synchronized time; the clock on each services PIC is autonomous (200–400 msec jitter) across PICs in a chassis
UNIX nanoseconds	Residual nanoseconds since 0000 UTC 1970	See Comments above for UNIX seconds
Flow sequence number	Sequence number of total flows received	–
Engine type	User-configured 8-bit value	Also known as VIP type on other vendors' equipment
Engine ID	User-configured 8-bit value	–

Figure 2: Version 5 Flow-Export Flow Header Format

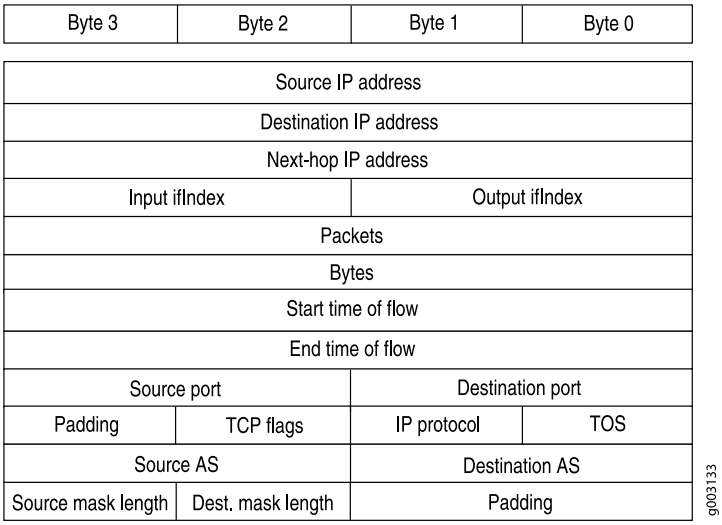


Table 4: Export Version 5 Flow-Export Flow Header Fields

Field	Description	Comments
Source IP address	Source IP address of the flow	–
Destination IP address	Destination IP address of the flow	–
Next-hop IP address	IP address of the router where flows are forwarded	–
Input ifIndex	SNMP index value for the input interface where the router receives flows	Junos OS Release 5.7 and later—Dynamically inserted, but overridden by manual configuration Junos OS Release 5.5—Manually set Junos OS Release 5.4—Set to zero
Output ifIndex	SNMP index value for the output interface where the router forwards flows	Junos OS Release 5.7 and later—Dynamically inserted, but overridden by manual configuration Junos OS Release 5.5—Manually set Junos OS Release 5.4—Set to zero
Packets	Total number of packets received in a flow	–
Bytes	Total number of bytes received in a flow	–

Table 4: Export Version 5 Flow-Export Flow Header Fields *(continued)*

Field	Description	Comments
Start time of flow	System up time, in seconds, at the start of the flow	System up time for the services PIC accepting flows
End time of flow	System up time, in seconds, at the end of the flow	System up time for the services PIC accepting flows
Source port	Source application port	–
Destination port	Destination application port	The ICMP type is placed in the high-order byte and the ICMP type code is placed in the low-order byte of this field
TCP flags	TCP flags set in the flow	–
IP protocol	IP protocol number	–
TOS	IP type of service	–
Source AS	AS number of the source address	Junos OS Release 5.7 and later—Dynamically inserted if AS information is available
Destination AS	AS number of the destination address	Junos OS Release 5.7 and later—Dynamically inserted if AS information is available
Source mask length	Source address network mask length	–
Dest. mask length	Destination address network mask length	–
Padding	Bytes available to ensure a minimum packet length	–

Useful formulas for flow monitoring are:

- start flow timestamp absolute = *unixTime* x 1000 – (*sysUptime* – start flow timestamp)
- end flow timestamp absolute = *unixTime* x 1000 – (*sysUptime* – end flow timestamp)

NOTE: In the 2-byte destination port field of the export version 5 flow-export flow format, the following information can be derived:

- High-order byte—ICMP type
- Low-order byte—ICMP type code

For example, if the ICMP type is 3 (00000011 in binary) and the ICMP type code is network unreachable (Type Code 0, or 00000000 in binary), the resulting destination port field value is 00000011 00000000 (768 in decimal).

For more information on ICMP type and type code, see RFC 792 at **http://www.ietf.org**.

Flow Monitoring Version 8 Format Output Fields

A detailed explanation of version 8 packet formats and fields is shown as follows:

Figure 3: Version 8 Template Flow Format

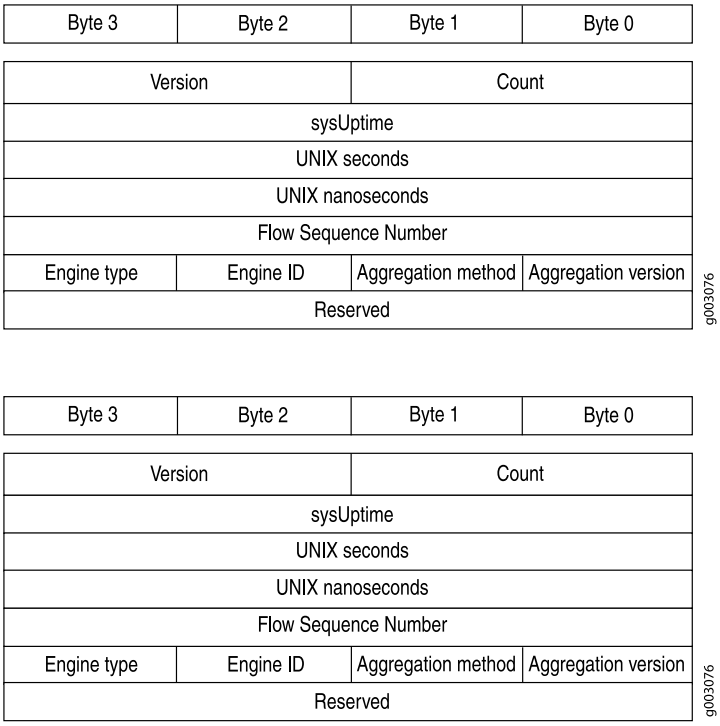


Table 5: Version 8 Flow Template Fields

Field	Description
Version	8
Count	The number of records in the protocol data unit (PDU) or packet
sysUptime	Current time elapsed, in milliseconds, since the router started
UNIX seconds	Current seconds since 0000 UTC 1970
UNIX nanoseconds	Residual nanoseconds since 0000 UTC 1970
Flow sequence number	Sequence counter of total flows received
Engine type	Type of flow switching engine
Engine ID	ID number of the flow switching engine
Aggregation method	Aggregation method used

Table 5: Version 8 Flow Template Fields *(continued)*

Field	Description
Aggregation version	Version of the aggregation export
Reserved	Empty field reserved for future usage

Figure 4: Version 8 AS Aggregation Flow Entry Format

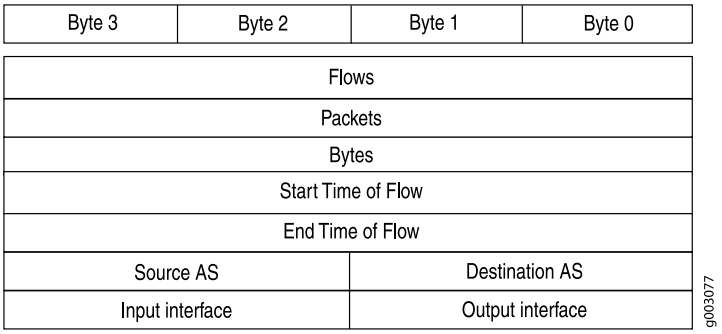


Table 6: Version 8 AS Aggregation Flow Entry Fields

Field	Description
Flows	Total number of flows
Packets	Total number of packets received in a flow
Bytes	Total number of bytes received in a flow
Start time of flow	System up time, in seconds, at the start of the flow
End time of flow	System up time, in seconds, at the end of the flow
Source AS	AS number of the source address
Destination AS	AS number of the destination address
Input interface	SNMP index value for the input interface where the router receives flows
Output interface	SNMP index value for the output interface where the router forwards flows

Figure 5: Version 8 Protocol/Port Aggregation Flow Entry Format

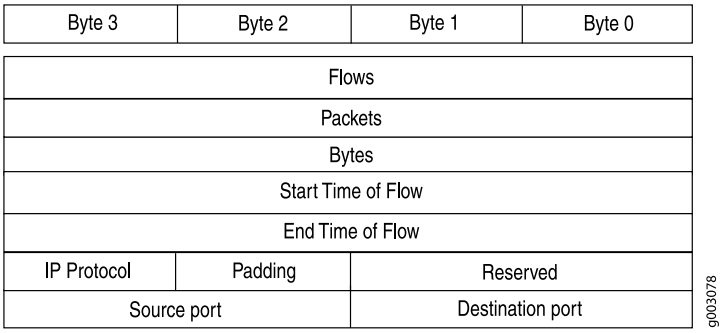


Table 7: Version 8 Protocol/Port Aggregation Flow Entry Fields

Field	Description
Flows	Total number of flows
Packets	Total number of packets received in a flow
Bytes	Total number of bytes received in a flow
Start time of flow	System up time, in seconds, at the start of the flow
End time of flow	System up time, in seconds, at the end of the flow
IP protocol	IP protocol number
Padding	Bytes available to ensure a minimum packet length
Reserved	Empty field reserved for future usage
Source port	Source application port
Destination port	Destination application port

Figure 6: Version 8 Prefix Aggregation Flow Entry Format

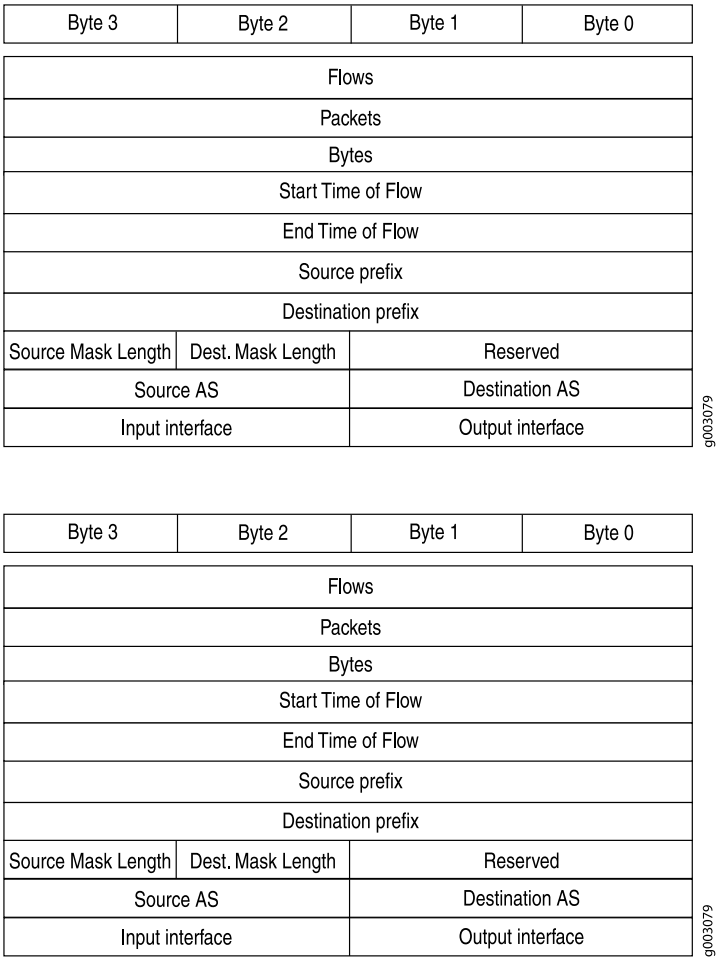


Table 8: Version 8 Prefix Aggregation Flow Entry Fields

Field	Description
Flows	Total number of flows
Packets	Total number of packets received in a flow
Bytes	Total number of bytes received in a flow
Start time of flow	System up time, in seconds, at the start of the flow
End time of flow	System up time, in seconds, at the end of the flow
Source prefix	Source IP address prefix

Table 8: Version 8 Prefix Aggregation Flow Entry Fields *(continued)*

Field	Description
Destination prefix	Destination IP address prefix
Source mask length	Source address network mask length
Dest. mask length	Destination address network mask length
Reserved	Empty field reserved for future usage
Source AS	AS number of the source address
Destination AS	AS number of the destination address
Input interface	SNMP index value for the input interface where the router receives flows
Output interface	SNMP index value for the output interface where the router forwards flows

Figure 7: Version 8 Source Prefix Aggregation Flow Entry Format

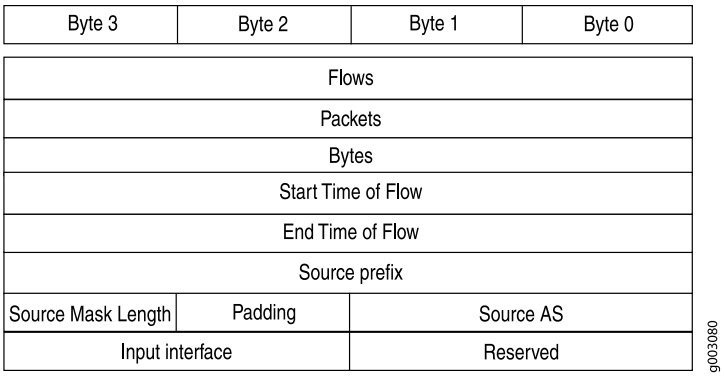


Table 9: Version 8 Source Prefix Aggregation Flow Entry Fields

Field	Description
Flows	Total number of flows
Packets	Total number of packets received in a flow
Bytes	Total number of bytes received in a flow

Table 9: Version 8 Source Prefix Aggregation Flow Entry Fields *(continued)*

Field	Description
Start time of flow	System up time, in seconds, at the start of the flow
End time of flow	System up time, in seconds, at the end of the flow
Source prefix	Source IP address prefix
Source mask length	Source address network mask length
Padding	Bytes available to ensure a minimum packet length
Source AS	AS number of the source address
Input interface	SNMP index value for the input interface where the router receives flows
Reserved	Empty field reserved for future usage

Figure 8: Version 8 Destination Prefix Aggregation Flow Entry Format

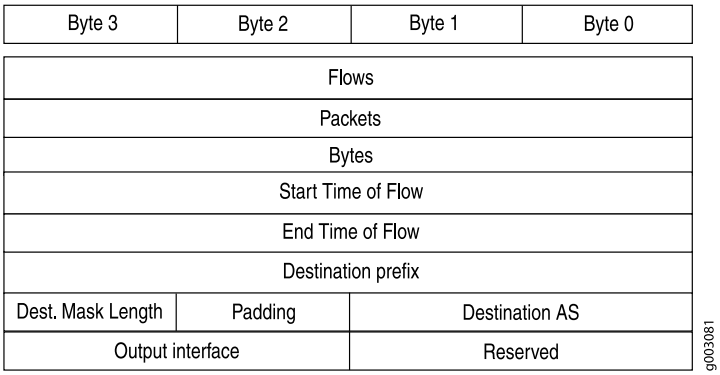


Table 10: Version 8 Destination Prefix Aggregation Flow Entry Fields

Field	Description
Flows	Total number of flows
Packets	Total number of packets received in a flow

Table 10: Version 8 Destination Prefix Aggregation Flow Entry Fields *(continued)*

Field	Description
Bytes	Total number of bytes received in a flow
Start time of flow	System up time, in seconds, at the start of the flow
End time of flow	System up time, in seconds, at the end of the flow
Destination prefix	Destination IP address prefix
Dest. mask length	Destination address network mask length
Padding	Bytes available to ensure a minimum packet length
Destination AS	AS number of the destination address
Output interface	SNMP index value for the output interface where the router forwards flows
Reserved	Empty field reserved for future usage

For more information about version 5 and version 8 packet formats and fields, see **http://www.caida.org**.

Flow Monitoring Version 9 Format Output Fields

IN THIS SECTION

A detailed explanation of active flow monitoring version 9 packet formats and fields is shown as follows:

- Table 11 on page 22
- Figure 9 on page 25

- Table 12 on page 25
- Figure 11 on page 28
- Table 12 on page 25
- Figure 12 on page 29
- Table 16 on page 29
- Figure 13 on page 30
- Table 17 on page 30

The Junos OS supports the version 9 template formats:

Table 11: Flow Monitoring Version 9 Template Formats

Template	Fields
IPv4	Flow selectors: • Source and destination IP address • Source and destination address prefix mask lengths • Source and destination port numbers • IP protocol and IP type of service • ICMP type Flow nonselectors: • TCP flags • Input and output SNMP • Input bytes • Input packets • Start time • End time

Table 11: Flow Monitoring Version 9 Template Formats *(continued)*

Template	Fields
MPLS	Flow selectors: • MPLS label 1 • MPLS label 2 • MPLS label 3 Flow nonselectors: • Input and output SNMP • Input bytes • Input packets • Start time • End time
MPLS_IPv4	Flow selectors: • MPLS label 1 • MPLS label 2 • MPLS label 3 • MPLS top-level FEC address Flow nonselectors: • Input and output SNMP • Input bytes • Input packets • Start time • End time

Table 11: Flow Monitoring Version 9 Template Formats *(continued)*

Template	Fields
IPv6	Flow selectors: • IP protocol and IP type of service • Source and destination port numbers • Input SNMP • Source and destination IPv6 address • ICMP type Flow nonselectors: • Input bytes • Input packets • TCP flags • Output SNMP • Source and destination autonomous system • Last and first switched • IPv6 source and destination mask • IP protocol version • IPv6 next hop
Peer AS billing	Flow selectors: • IPv4 class of service • Ingress interface information • BGP peer destination AS number • BGP IPv4 nest hop address Flow nonselectors • Input and output SNMP • Input bytes • Input packets • First switch • Last switched **NOTE:** Peer AS billing traffic is not supported for active flow monitoring version 9 configuration on PTX5000 routers tethered to CSE2000.

Figure 9: Version 9 Flow Header Format

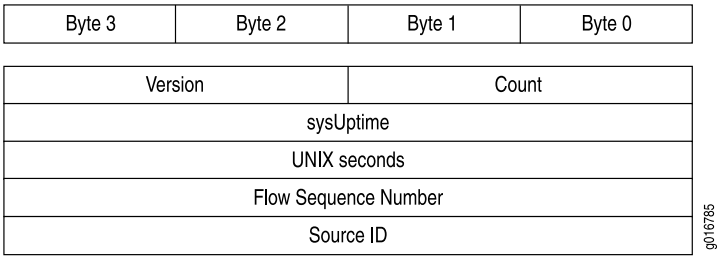


Table 12: Version 9 Flow Header Fields

Field	Description
Version	9
Count	Total number of records in the protocol data unit (PDU) or packet. This number includes all of the options FlowSet records, template FlowSet records, and data FlowSet records.
sysUptime	Current time elapsed, in milliseconds, since the router started.
UNIX seconds	Current seconds since 0000 UTC 1970.
Flow sequence number	Sequence counter of total flows received.
Source ID	32-bit value that identifies the data exporter. Version 9 uses the integrated field diagnostics (IFD) SNMP index of the PIC or device that is exporting the data flow. This field is equivalent to engine type and engine ID fields found in versions 5 and 8.

Figure 10: Version 9 Template FlowSet Format

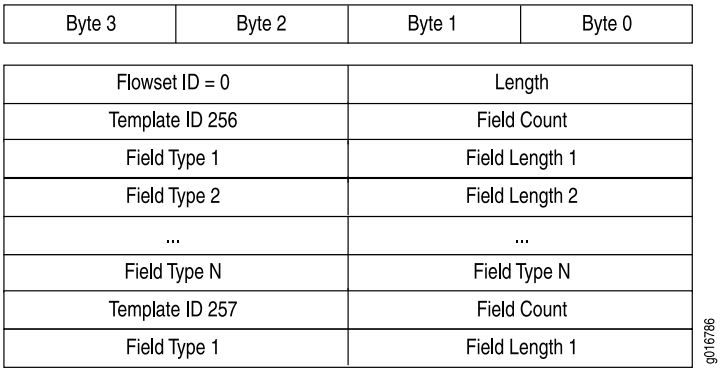


Table 13: Version 9 Template FlowSet Fields

Field	Description
FlowSet ID	FlowSet type. FlowSet ID 0 is reserved for the Template FlowSet.
Length	FlowSet length. Individual template FlowSets might contain multiple template records, which means that the length of template FlowSets varies.
Template ID	Unique template ID assigned to each newly generated template. Templates numbered 256 and higher define data formats. Templates numbered 0 through 255 define FlowSet IDs.
Field Count	Fields in the template record. This field allows the collector to determine the end of the current template record and the start of the next.
Field Type	Field type. These are defined in Table 14 on page 26.
Field Length	Length, in bytes, of the corresponding field type.

Table 14: Field Type Definitions Supported in Junos OS

Field Type	Description
1	IN_BYTES: The number of bytes associated with an IP flow. By default, the length is 4 bytes.

Table 14: Field Type Definitions Supported in Junos OS *(continued)*

Field Type	Description
2	IN_PKTS: The number of packets associated with an IP flow. By default, the length is 4 packets.
4	PROTOCOL: The IP protocol byte.
5	TOS: The type-of-service byte setting of an incoming packet.
6	TCP_FLAGS: The cumulative TCP flags associated with a flow.
7	L4_SRC_PORT: The TCP/UDP source port.
8	IPv4_SRC_ADDR: The IPv4 source address.
9	SRC_MASK: The number of contiguous bits in the source subnet mask.
10	INPUT_SNMP: The IFD SNMP input interface index. By default, the length is 2.
11	L4_DST_PORT: The TCP/UDP destination port number.
12	IPV4_DST_ADDR: The IPv4 destination address.
13	DST_MASK: The number of contiguous bits in the destination subnet mask.
14	OUTPUT_SNMP: The IFD SNMP output interface index. By default, the length is 2.
16	SRC_AS: The source autonomous system number. This is always set to zero.
17	DST_AS: The destination autonomous system number. This is always set to zero.
18	BGP_IPV4_NEXT_HOP: The BGP IPV4 next-hop address.
21	LAST_SWITCHED: The uptime of the device (in milliseconds) at which the last packet of the flow was switched.
22	FIRST_SWITCHED: The uptime of the device (in milliseconds) at which the first packet of the flow was switched.
29	IPV6_SRC_MASK: The length of the IPv6 source mask, in contiguous bits.

Table 14: Field Type Definitions Supported in Junos OS *(continued)*

Field Type	Description
30	IPV6_DST_MASK: The length of the IPv6 destination mask, in contiguous bits.
32	ICMP_TYPE: The ICMP type.
34	SAMPLING_INTERVAL: The rate at which packets are sampled. As an example, a rate of 100 means that one packet is sampled for every 100 packets in the data flow.
35	SAMPLING_ALGORITHM: The type of algorithm being used. Ox01 indicates deterministic sampling and 0x02 indicates random sampling.
47	MPLS_TOP_LABEL_IP_ADDRESS: The MPLS top- label address.
60	IP_PROTOCOL_VERSION: The IP protocol version being used.
62	IPV6_NEXT_HOP: The IPv6 address of the next-hop router.
70	MPLS_LABEL_1: The first MPLS label in the stack.
71	MPLS_LABEL_2: The second MPLS label in the stack.
72	MPLS_LABEL_3: The third MPLS label in the stack.
128	DST_PEER_AS: The destination of the BGP peer AS.

Figure 11: Version 9 Data FlowSet Format

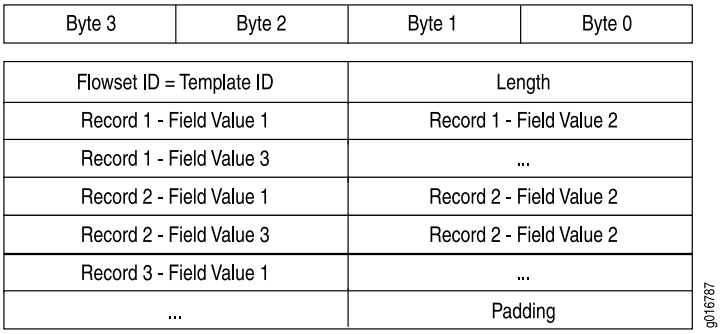


Table 15: Version 9 Data FlowSet Format

Field	Description
FlowSet ID = Template ID	Data FlowSet that associated with a FlowSet ID. The FlowSet ID maps to a previously generated template ID. The flow collector must use the FlowSet ID to find the corresponding template record and decode the flow records from the FlowSet.
Length	FlowSet length. Data FlowSets are fixed in length.
Record Number - Field Value Number	Flow data records, each containing a set of field values. The template record identified by the FlowSet ID dictates the type and length of the field values.
Padding	Bytes (in zeros) that the exporter inserts so that the subsequent FlowSet starts at a 4-byte aligned boundary.

Figure 12: Version 9 Options Template Format

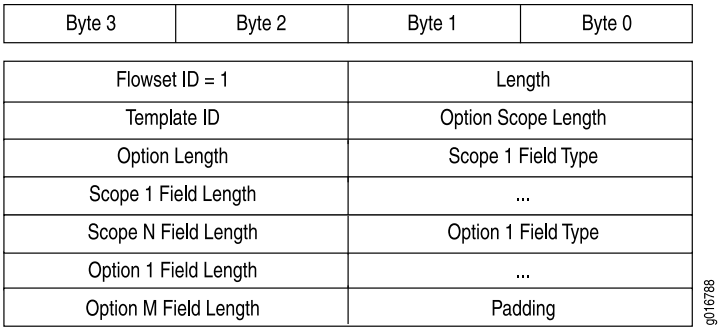


Table 16: Version 9 Options Template Format

Field	Description
FlowSet ID	FlowSet type. FlowSet ID 1 is reserved for the options template.
Length	FlowSet length. Option template FlowSets are fixed in length.

Table 16: Version 9 Options Template Format *(continued)*

Field	Description
Template ID	Template ID of the options template. Options template values are greater than 255.
Option Scope Length	Length, in bytes, of any scope field definition that is part of the options template record.
Scope 1 Field Type	Relevant process. The Junos OS supports the system process (1).
Scope 1 Field Length	Length, in bytes, of the option field.
Padding	Bytes the exporter inserts so that the subsequent FlowSet starts at a 4-byte aligned boundary.

Figure 13: Active Flow Monitoring Version 9 Options Data Record Format

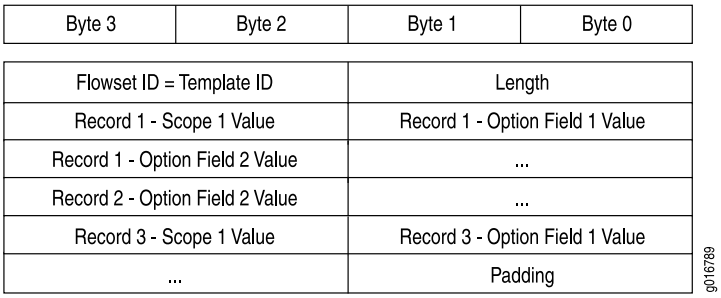


Table 17: Active Flow Monitoring Version 9 Options Data Record Format

Field	Description
FlowSet ID = Template ID	ID that precedes each options data flow record. The FlowSet ID maps to a previously generated template ID. The collector must use the FlowSet ID to find the corresponding template record and decode the options data flow records from the FlowSet.
Length	FlowSet length. Option FlowSets are fixed in length.

Table 17: Active Flow Monitoring Version 9 Options Data Record Format *(continued)*

Field	Description
Number of Flow Data Records	Remainder of the options data FlowSet is a collection of flow data records, each containing a set of field values. The template record identified by the FlowSet ID dictates the type and length of the field values.
Padding	Bytes (in zeros) the exporter inserts so that the subsequent FlowSet starts at a 4-byte aligned boundary.

IPFIX (Version 10) IPv4 Fields

Field Name	Flow Key	Element ID	Length in Bytes
IPV4_SADDR	Y	8	4
IPV4_DADDR	Y	12	4
IPV4_TOS	Y	5	1
IPV4_PROTO	Y	4	1
TCP_UDP_SPORT	Y	7	2
TCP_UDP_DPORT	Y	11	2
IMCP_TYPE_CODE_IPV4	Y	32	2
IIF	Y	10	4
VLAN_ID	Configurable	58	2
IPV4_SMASK	N	9	1
IPV4_DMASK	N	13	1
SRC_AS	N	16	4

Field Name	Flow Key	Element ID	Length in Bytes
DST_AS	N	17	4
IPV4_NEXTHOP	N	15	4
TCP_FLAGS	N	6	1
OIF	N	14	4
FLOW_BYTES	N	1	8
FLOW_PACKETS	N	2	8
MIN_TTL	N	52	1
MAX_TTL	N	53	1
START_TIME	N	152	8
END_TIME	N	153	8
FIRST_SWITCHED	N	22	4
LAST_SWITCHED	N	21	4
FLOW_END_REASON	N	136	1
IP_PROTOCOL_VERSION	N	60	1
BGP_NEXTHOP_ID	N	18	4
FLOW_DIRECTION	Configurable	61	1
DOT_1Q_VLAN_ID	N	243	2
DOt_1Q_CUSTOMER_VLAN_ID	N	245	2
IP IDENTIFIER	N	54	4

CHAPTER 2

Monitoring Traffic Using Active Flow Monitoring

IN THIS CHAPTER

Configuring Active Flow Monitoring

In active flow monitoring, the router participates in both the monitoring application and in the normal routing functionality of the network. Although the Monitoring Services PIC was designed initially for use as an offline passive flow monitoring tool, it can also be used in an active flow monitoring topology.

Table 18 on page 35 shows which Juniper Networks PICs and corresponding routers support active flow monitoring. For more information on Juniper Networks PICs, see the PIC guide that corresponds to your router.

Table 18: Passive and Active Flow Monitoring PIC Support

PIC Type and Service	M5/M10	M7i/M10i	M20	M40e	M120	M160	T Series/ M320	TX Matrix
Monitoring Services PIC: active flow monitoring	Yes (version 8 only)	Yes	Yes	Yes	No	Yes (version 8 only)	No	No
Monitoring Services II PIC: flow collection services	No	No	No	Yes	No	Yes (version 8 only)	No	No
Adaptive Services PIC: active flow monitoring	Yes (version 8 only)	Yes	Yes	Yes	No	Yes (version 8 only)	No	No
Adaptive Services II PIC: active flow monitoring	Yes (version 8 only)	Yes	Yes	Yes	Yes	Yes (version 8 only)	Yes	Yes
Adaptive Services II PIC: flow-tap services	No	Yes	Yes	Yes	Yes	No	Yes	No
MultiServices 100 PIC: active flow monitoring	No	Yes	No	Yes	No	No	Yes	Yes
MultiServices 400 PIC: active flow monitoring	No	No	No	Yes	Yes	No	Yes	Yes
MultiServices 500 PIC: active flow monitoring	No	No	No	Yes	Yes	No	Yes	Yes

Table 18: Passive and Active Flow Monitoring PIC Support *(continued)*

PIC Type and Service	M5/M10	M7i/M10i	M20	M40e	M120	M160	T Series/ M320	TX Matrix
Junos OS-enabled active flow monitoring	No	No	No	No	No	No	No	No

Specified packets can be filtered and sent to the monitoring interface. For the Monitoring Services PIC, the interface name contains the **mo-** prefix. For the Adaptive Services PICs and MultiServices PICs, the interface name contains the **sp-** prefix.

> **NOTE:** If you upgrade from the Monitoring Services PIC to the Adaptive Services PIC or MultiServices PIC for active flow monitoring, you must modify the interface name of your monitoring interface from **mo-*fpc/pic/port*** to **sp-*fpc/pic/port***.

The major active flow monitoring actions you can configure at the **[edit forwarding-options]** hierarchy level are as follows:

- Sampling, with the [**edit forwarding-options sampling**] hierarchy. This option extracts limited information (such as the source and destination IP address) from a copy of some of the packets in a flow, while the original packets are forwarded to the intended destination. This option is extended to define active sampling on a per Packet Forwarding Engine basis by defining a sampling instance that specifies a name for the sampling parameters and binding the instance to the particular Packet Forwarding Engine.
- Templates, with the [**edit forwarding-options sampling**] and [**edit services monitoring**] hierarchies. With active flow monitoring support for version 5, version 8, and the customizing version 9, you can use templates to organize the data gathered from sampling.
- Discard accounting, with the [**edit forwarding-options accounting**] hierarchy. This option quarantines unwanted packets, creates flow monitoring records that describe the packets, and discards the packets instead of forwarding them.
- Port mirroring, with the [**edit forwarding-options port-mirroring**] hierarchy. This option makes one full copy of all packets in a flow and delivers the copy to a single destination.
- Multiple port mirroring, with the [**edit forwarding-options next-hop-group**] hierarchy. This option allows multiple copies of selected traffic to be delivered to multiple destinations. (Multiple port mirroring requires a Tunnel Services PIC.)
- Flow-tap services processing, with the [**edit services flow-tap**] hierarchy. This option sends copies of packets that match dynamic filter criteria to one or more content destinations.

Unlike passive flow monitoring, you do not need to configure a monitoring group. Instead, you can send filtered packets to a monitoring services or adaptive services interface (**mo-** or **sp-)** by using sampling or discard accounting. Optionally, you can configure port mirroring or multiple port mirroring to direct packets to additional interfaces.

These active flow monitoring options provide a wide variety of actions that can be performed on network traffic flows. However, the following restrictions apply:

- The router can perform either sampling *or* port mirroring at any one time.
- The router can perform either forwarding *or* discard accounting at any one time.

Because the Monitoring Services PIC, Adaptive Services PIC, and MultiServices PIC allow only one action to be performed at any one time, the following configuration options are available:

- Sampling and forwarding
- Sampling and discard accounting
- Port mirroring and forwarding
- Port mirroring and discard accounting
- Sampling and port mirroring on different sets of traffic

To configure active flow monitoring, complete these steps:

- Defining a Firewall Filter on M, MX and T Series Routers to Select Traffic for Active Flow Monitoring on page 81
- Configuring Actively Monitored Interfaces on M, MX and T Series Routers on page 76
- Processing IPv4 traffic on an M, MX or T Series Router Using Monitoring services, Adaptive services or Multiservices Interfaces on page 82
- Collecting Flow Records on page 76
- Rerouting Packets on an M, MX or T Series Router with Port Mirroring on page 170
- Option: Configuring Port Mirroring with Filter-Based Forwarding and a Monitoring Group on page 111
- Sending Port-Mirrored Traffic from an M, MX or T Series Router to Multiple Export Interfaces by Using Next-Hop Groups on page 111
- Sending Packets to a Mediation Device on MX, M and T Series Routers on page 340

Active Flow Monitoring System Requirements

To implement active flow monitoring, your system must meet these minimum requirements:

- Junos 10.4 or later for peer AS billing support on flow monitoring version 9
- Junos 9.3R2 or later for IPv6 support on flow monitoring version 9
- Junos 9.3R2 or later for multiple flows for flow monitoring version 9
- Junos OS Release 9.0 or later for version 9 flow aggregation to multiple flow servers
- Junos OS Release 8.5 or later for active flow monitoring support on MultiServices 500 PICs
- Junos OS Release 8.3 or later for flow monitoring version 9 support, MPLS support, and active flow monitoring support on MultiServices 100 and 400 PICs
- Junos OS Release 8.2 or later for M120 router support and for flow monitoring version 5 and 8 support on MultiServices 100 and 400 PICs
- Junos OS Release 8.1 or later for the flow-tap services application on Adaptive Services II PICs installed in M7i, M10i, M20, M40e, M320, and T Series routers
- Junos OS Release 7.4 or later for port mirroring of IPv6 packets
- Junos OS Release 7.3 or later for active flow monitoring on Adaptive Services II PICs installed in TX Matrix platforms
- Junos OS Release 7.0 or later for active flow monitoring on Adaptive Services II PICs installed in T Series and M320 routers
- Junos OS Release 6.0 or later for the Adaptive Services PIC
- Junos OS Release 5.7 or later for the automatic insertion of AS numbers and SNMP index values for input and output interfaces into records, port mirroring to multiple ports, and discard accounting
- Junos OS Release 5.6 or later for the Monitoring Services PIC
- M5, M7i, M10, M10i, M20, M40e, M120, M160, M320, or T Series router with an Internet Processor II ASIC or later
- Type 1 enhanced FPCs
- Two M Series or T Series PICs of your choice: One to receive incoming traffic and one to forward outgoing traffic (the second PIC or PIM is not necessary for discard accounting)
- Export PICs to connect to the collector or packet analyzer
- Tunnel Services PIC (required for multiple port mirroring or **mo-** interface load balancing)
- Flow collector version 5, 8, or 9
- ES PIC and packet analyzers (optional)

RELATED DOCUMENTATION

Passive Flow Monitoring System Requirements for T Series, M Series and MX Series Routers | 200

Active Flow Monitoring PIC Specifications | 40

Active Flow Monitoring Applications

Flow monitoring can be used for many different reasons such as network planning, accounting, usage-based network billing, security, and monitoring for Denial-of-Service attacks.

Some examples of the types of things you can use flow monitoring for are:

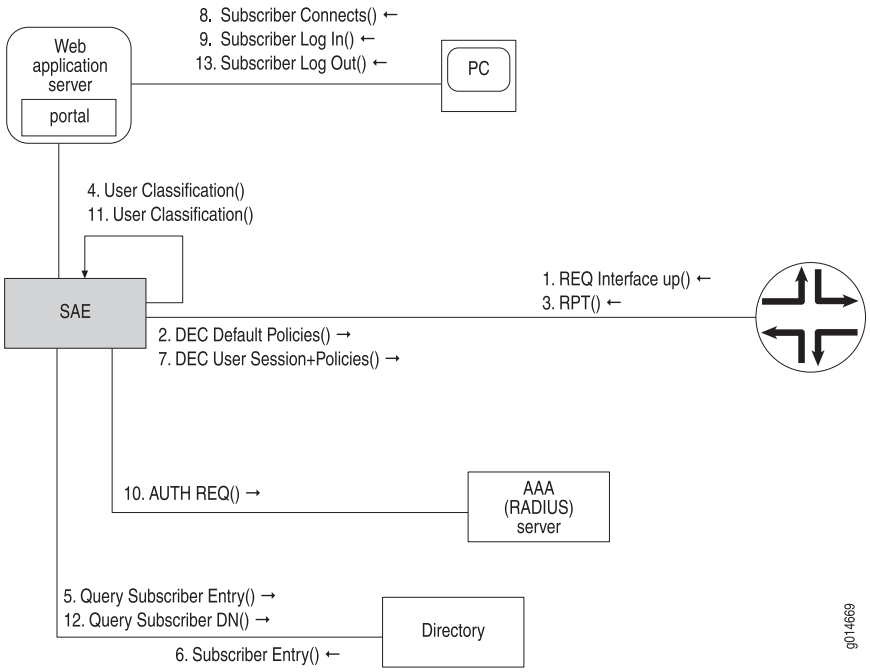
- Tracking what kind of traffic is entering or exiting an ISP or corporate network.
- Tracking traffic flows between BGP autonomous systems.
- Tracking traffic flows between enterprise network regions.
- Taking a snapshot of the existing quality-of-service (QoS) policy results prior to making changes in QoS policy in case you need to roll back changes later in the process.
- Verifying that load balancing techniques are performing as intended.
- Capturing a base line of current network performance prior to making changes intended to improve performance so that you know if the changes are helping.
- Discovering if network users at an enterprise are using bandwidth for work-related activities or for non work-related activities.

Examples of how flow monitoring helps with network administration include the following:

- A large service provider uses active flow monitoring on its core uplinks as a way to collect data on the protocols in use, packet sizes, and flow durations to better understand the usage of its Internet service offering. This helps the provider understand where network growth is coming from.
- Service providers bill customers for the data sent or bandwidth used by sending captured flow data to third-party billing software.
- At a large enterprise, VoIP users at aremote site complained of poor voice quality. The flow monitoring reports showed that the VoIP traffic did not have the correct type of service settings.
- Users on an enterprise network, reported network slowdowns. The flow monitoring reports showed that one user's PC was generating a large portion of the network traffic. The PC was infected with malware.
- A growing enterprise planned to deploy new business management software and needed to know what type of network bandwidth demand the new software would create. During the software trial period, flow monitoring reports were used to identify the expected increase in traffic.

Thus, while flow monitoring is traditionally associated with traffic analysis, it also has a role in accounting and security.

Figure 14: Active Flow Monitoring



RELATED DOCUMENTATION

Flow Monitoring Overview

Active Flow Monitoring Overview | 44

Active Flow Monitoring PIC Specifications

For Monitoring Services PIC specifications, see Table 19 on page 40 and Table 20 on page 41. For Adaptive Services PIC specifications, see Table 21 on page 42. For MultiServices PIC specifications, see Table 22 on page 42 and Table 23 on page 43.

Table 19: Monitoring Services PIC Specifications

Specification	Description
Physical dimensions	Single-wide PIC that occupies one PIC slot

Table 19: Monitoring Services PIC Specifications *(continued)*

Specification	Description
Connectors	DB-9 diagnostic serial console port
Status LED	One tricolor: • Off—The PIC is offline; it is safe to remove it from the chassis. • Green—The PIC is operating normally. • Amber—The PIC is initializing. • Red—The PIC has an error or failure; no further harm can be done by removing it from the chassis.
Application LED	One tricolor: • Off—The service is not running. • Green—The service is running under acceptable load. • Amber—The service is overloaded.

Table 20: Monitoring Services II PIC Specifications

Specification	Description
Physical dimensions	Single-wide PIC that occupies one PIC slot
Connectors	N/A
Status LED	One tricolor: • Off—The PIC is offline; it is safe to remove it from the chassis. • Green—The PIC is operating normally. • Amber—The PIC is initializing. • Red—The PIC has an error or failure; no further harm can be done by removing it from the chassis.
Application LED	One tricolor: • Off—The flow collector is not running. • Green—The flow collector is running under acceptable load. • Amber—The flow collector is overloaded.

Table 21: Adaptive Services PIC Specifications

Specification	Description
Physical dimensions	Single-wide PIC that occupies one PIC slot
Connectors	N/A
Status LED	One tricolor: • Off—The PIC is offline; it is safe to remove it from the chassis. • Green—The PIC is operating normally. • Amber—The PIC is initializing. • Red—The PIC has an error or failure; no further harm can be done by removing it from the chassis.
Application LED	One tricolor: • Off—The flow collector is not running. • Green—The flow collector is running under acceptable load. • Amber—The flow collector is overloaded.

Table 22: MultiServices 100 PIC

Specification	Description
Physical dimensions	Single-wide PIC that occupies one PIC slot
Connectors	N/A
Status LED	One tricolor: • Off—The PIC is offline; it is safe to remove it from the chassis. • Green—The PIC is operating normally. • Amber—The PIC is initializing. • Red—The PIC has an error or failure; no further harm can be done by removing it from the chassis.
Application LED	One tricolor: • Off—The service is not running. • Green—The service is running under acceptable load. • Amber—The service is overloaded.

Table 23: MultiServices 400 PIC

Specification	Description
Physical dimensions	Single-wide PIC that occupies one PIC slot
Connectors	N/A
Status LED	One tricolor: • Off—The PIC is offline; it is safe to remove it from the chassis. • Green—The PIC is operating normally. • Amber—The PIC is initializing. • Red—The PIC has an error or failure; no further harm can be done by removing it from the chassis.
Application LED	One tricolor: • Off—The service is not running. • Green—The service is running under acceptable load. • Amber—The service is overloaded.

Table 24: MultiServices 500 PIC

Specification	Description
Physical dimensions	Single-wide PIC that occupies one PIC slot
Connectors	N/A
Status LED	One tricolor: • Off—The PIC is offline; it is safe to remove it from the chassis. • Green—The PIC is operating normally. • Amber—The PIC is initializing. • Red—The PIC has an error or failure; no further harm can be done by removing it from the chassis.
Application LED	One tricolor: • Off—The service is not running. • Green—The service is running under acceptable load. • Amber—The service is overloaded.

RELATED DOCUMENTATION

Passive Flow Monitoring System Requirements for T Series, M Series and MX Series Routers | 200

Active Flow Monitoring System Requirements | 38

Active Flow Monitoring Overview

Flow monitoring versions 5, 8, and 9 support active flow monitoring. For active flow monitoring, the monitoring station participates in the network as an active router. The major actions the router can perform during active flow monitoring are as follows:

- Sampling—The router selects and analyzes only a portion of the traffic.
- Sampling with templates—The router selects, analyzes, and arranges a portion of the traffic into templates.
- Sampling per sampling instance—The router selects, analyzes, and arranges a portion of the traffic according to the configuration and binding of a sampling instance.
- Port mirroring—The router copies entire packets and sends the copies to another interface.
- Multiple port mirroring—The router sends multiple copies of monitored packets to multiple export interfaces with the **next-hop-group** statement at the **[edit forwarding-options]** hierarchy level.
- Discard accounting—The router accounts for selected traffic before discarding it. Such traffic is not forwarded out of the router. Instead, the traffic is quarantined and deleted.
- Flow-tap processing—The router processes requests for active flow monitoring dynamically by using the Dynamic Tasking Control Protocol (DTCP).

RELATED DOCUMENTATION

Flow Monitoring Overview

Understanding Passive Flow Monitoring on T Series, M Series and MX Series Routers | 197

Active Flow Monitoring Overview

Using a Juniper Networks M Series Multiservice Edge or T Series Core router or EX9200, a selection of PICs (including the Monitoring Services PIC, Adaptive Services [AS] PIC, Multiservices PIC, or Multiservices DPC) and other networking hardware, you can monitor traffic flow and export the monitored traffic. Monitoring traffic allows you to do the following:

- Gather and export detailed information about IP version 4 (IPv4) traffic flows between source and destination nodes in your network.
- Sample all incoming IPv4 traffic on the monitoring interface and present the data in cflowd record format.
- Perform discard accounting on an incoming traffic flow.
- Encrypt or tunnel outgoing cflowd records, intercepted IPv4 traffic, or both.
- Direct filtered traffic to different packet analyzers and present the data in its original format (port mirror).

> **NOTE:** Monitoring Services PICs, AS PICs, and Multiservices PICs must be mounted on an Enhanced Flexible PIC Concentrator (FPC) in an M Series or T Series router.
>
> Multiservices DPCs installed in Juniper Networks MX Series 3D Universal Edge routers support the same functionality, with the exception of the passive monitoring and flow-tap features.

Although the Monitoring Services PIC was designed initially for use as an offline passive flow monitoring tool, it can also be used in an active flow monitoring topology. In contrast, the AS or Multiservices PIC is designed exclusively for active flow monitoring. To use either the Monitoring Services PIC, AS PIC, or Multiservices PIC for active flow monitoring, you must install the PIC in an M Series or T Series router. The router participates in both the monitoring application and in the normal routing functionality of the network.

Starting with Junos OS Release 11.4, support for active monitoring is extended to logical systems running on T Series and MX Series routers. A logical system is a partition created from a physical router that performs independent routing tasks. Several logical systems in a single router with their own interfaces, policies, instances, and routing tables can perform functions handled by several different routers. A shared services PIC handles flows from all the logical systems. Only version 9 flows, IPv4, and MPLS templates are supported. See “Example: Configuring Active Monitoring on an M, MX or T Series Router’s Logical System” on page 48 for a sample configuration that enables active monitoring on a logical system.

Specified packets can be filtered and sent to the monitoring interface. For the Monitoring Services PIC, the interface name contains the **mo-** prefix. For the AS or Multiservices PIC, the interface name contains the **sp-** prefix.

> **NOTE:** If you upgrade from the Monitoring Services PIC to the Adaptive Services or Multiservices PIC for active flow monitoring, you must change the name of your monitoring interface from mo-*fpc/pic/port* to sp-*fpc/pic/port*.

The major active flow monitoring actions you can configure at the **[edit forwarding-options]** hierarchy level are as follows:

- Sampling, with the **[edit forwarding-options sampling**] hierarchy. This option sends a copy of the traffic stream to an AS or Monitoring Services PIC, which extracts limited information (such as the source and destination IP address) from some of the packets in a flow. The original packets are forwarded to the intended destination as usual.
- Discard accounting, with the **[edit forwarding-options accounting]** hierarchy. This option quarantines unwanted packets, creates cflowd records that describe the packets, and discards the packets instead of forwarding them.
- Port mirroring, with the **[edit forwarding-options port-mirroring]** hierarchy. This option makes one full copy of all packets in a flow and delivers the copy to a single destination. The original packets are forwarded to the intended destination.
- Multiple port mirroring, with the **[edit forwarding-options next-hop-group]** hierarchy. This option allows multiple copies of selected traffic to be delivered to multiple destinations. (Multiple port mirroring requires a Tunnel Services PIC.)

Unlike passive flow monitoring, you do not need to configure a monitoring group. Instead, you can send filtered packets to a monitoring services or adaptive services interface (**mo-** or **sp-**) by using sampling or discard accounting. Optionally, you can configure port mirroring or multiple port mirroring to direct packets to additional interfaces.

These active flow monitoring options provide a wide variety of actions that can be performed on network traffic flows. However, the following restrictions apply:

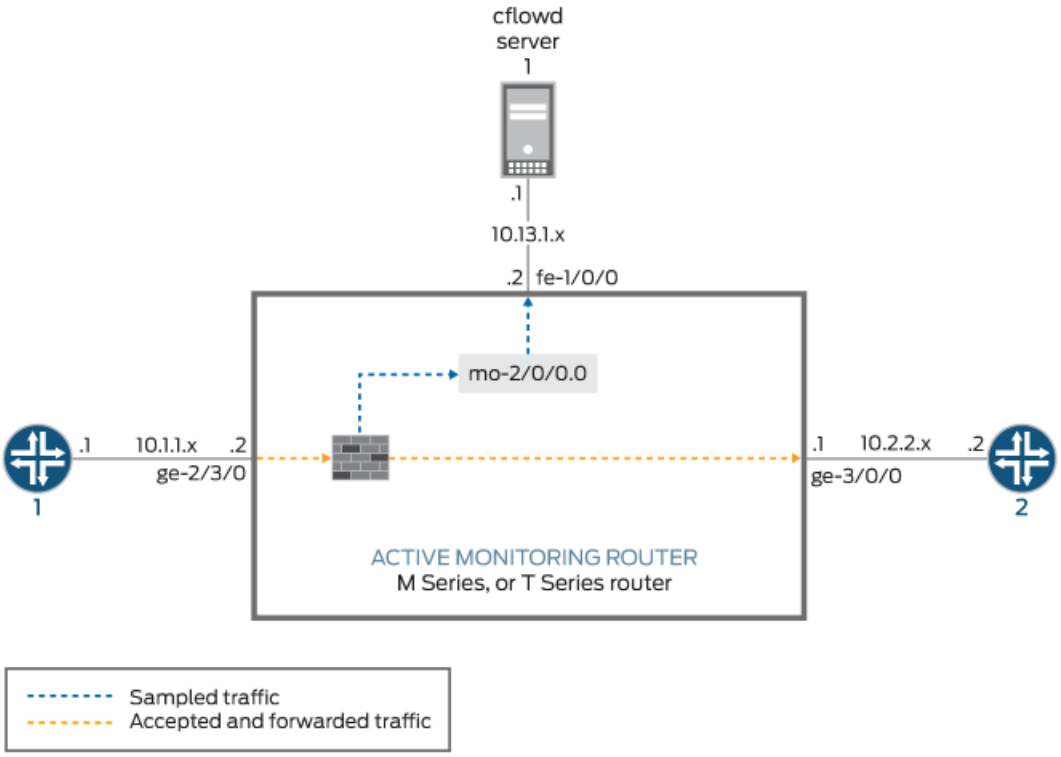
- The router or switch can perform sampling *or* port mirroring at any one time.
- The router or switch can perform forwarding *or* discard accounting at any one time.

Because the Monitoring Services, AS, and Multiservices PICs allow only one action to be performed at any one time, the following configuration options are available:

- Sampling and forwarding
- Sampling and discard accounting
- Port mirroring and forwarding
- Port mirroring and discard accounting
- Sampling and port mirroring on different sets of traffic

Figure 15 on page 47 shows a sample topology.

Figure 15: Active Monitoring Configuration Topology



In Figure 15 on page 47, traffic from Router 1 arrives on the monitoring router's Gigabit Ethernet ge-2/3/0 interface. The exit interface on the monitoring router leading to destination Router 2 is ge-3/0/0, but this can be any interface type (such as SONET, Gigabit Ethernet, and so on). The export interface leading to the cflowd server is fe-1/0/0.

To enable active monitoring, configure a firewall filter on the interface ge-2/3/0 with the following match conditions:

- Traffic matching certain firewall conditions is sent to the Monitoring Services PIC using filter-based forwarding. This traffic is quarantined and not forwarded to other routers.
- All other traffic is port-mirrored to the Monitoring Services PIC. Port mirroring copies each packet and sends the copies to the port-mirroring next hop (in this case, a Monitoring Services PIC). The original packets are forwarded out of the router as usual.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | 540

Configuring Services Interface Redundancy with Flow Monitoring | 61

Example: Configuring Active Monitoring on an M, MX or T Series Router's Logical System | 48

Example: Configuring Active Monitoring on an M, MX or T Series Router's Logical System

This example shows a sample configuration that allows you to configure active monitoring on a logical M-series, MX-series, T-series, or PTX Series system.

The following section shows the configuration on the master router:

```
[edit forwarding-options]
sampling {
    instance inst1 {
        input {
            rate 1;
        }
        family inet;
            output {
                flow-server 198.51.100.2 {
                    port 2055;
                    version9 {
                        template {
                            ipv4;
                        }
                    }
                }
            }
            interface sp-0/1/0 {
                source-address 10.11.12.13;
            }
        }
    }
    family mpls;
        output {
            flow-server 198.51.100.2 {
                port 2055;
```

```
            version9 {
               template {
                  mpls;
               }
            }
         }
      }
      interface sp-0/1/0 {
         source-address 10.11.12.13;
      }
   }
}
services {
   flow-monitoring {
      version9 {
         template ipv4 {
            flow-active-timeout 60;
            flow-inactive-timeout 60;
            ipv4-template;
            template-refresh-rate {
               packets 1000;
               seconds 10;
            }
            option-refresh-rate {
               packets 1000;
               seconds 10;
            }
         }
         template mpls {
            mpls-template;
         }
      }
   }
}
```

The configuration for the logical router uses the input parameters and the output interface for sampling from the master router. Each logical router should have separate template definitions for the flow-server configuration. The following section shows the configuration on the logical router:

```
logical-systems {
   ls-1 {
      firewall {
         family inet {
            filter test-sample {
```

```
            term term-1 {
                then {
                    sample;
                    accept;
                }
            }
        }
    }
}
interfaces {
    ge-0/0/1 {
        unit 0 {
            family inet {
                filter {
                    input test-sample;
                    output test-sample;
                }
            }
        }
    }
}
forwarding-options {
    sampling {
        instance sample-inst1 {
            family inet;
                output {
                    flow-server 198.51.100.2 {
                        port 2055;
                        version9 {
                            template {
                                ipv4-ls1;
                            }
                        }
                    }
                }
            }
        }
        family mpls;
            output {
                flow-server 198.51.100.2 {
                    port 2055;
                    version9 {
                        template {
                            mpls-ls1;
```

```
                    }
                  }
                }
              }
            }
          }
        }
        services {
          flow-monitoring {
            version9 {
              template ipv4-ls1 {
                flow-active-timeout 60;
                flow-inactive-timeout 60;
                ipv4-template;
                template-refresh-rate {
                  packets 1000;
                  seconds 10;
                }
                option-refresh-rate {
                  packets 1000;
                  seconds 10;
                }
              }
              template mpls-ls1 {
                mpls-template;
              }
            }
          }
        }
      }
    }
```

RELATED DOCUMENTATION

- Active Flow Monitoring Overview | 45
- Configuring Flow Monitoring | 3
- Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | 540
- Configuring Services Interface Redundancy with Flow Monitoring | 61

Example: Configuring Flow Monitoring on an MX Series Router with MS-MIC and MS-MPC

IN THIS SECTION

This example shows how you can configure Junos Traffic Vision for flow monitoring on an MX Series Router with MS-MIC and MS-MPC, and contains the following sections:

Hardware and Software Requirements

This example requires an MX Series router that has:

- Junos OS Release 13.2 running on it.
- An MS-MIC installed in it.

Junos Traffic Vision Support on MS-MIC and MS-MPC

Junos Traffic Vision (previously known as Jflow) is the accounting service that is available on the MS-MIC and MS-MPC. Junos Traffic Vision enables users to keep track of the packets received on the MS-MIC or MS-MPC and to generate flow records that contain information such as the source address of the packet, the destination address of the packet, packets and byte counts, and so on. Junos Traffic Vision implementation does not interrupt the traffic, instead it makes a copy of the incoming packet and sends that copy to the service interface card for analyzing the information and maintaining the record.

Starting with Release 13.2, the Junos OS extension-provider packages come preinstalled on a multiservices MIC and MPC (MS-MIC and MS-MPC). The **adaptive-services** configuration at the **[edit chassis fpc *number* pic *number*]** hierarchy level is preconfigured on these cards.

Before you configure Junos Traffic Vision on an MS-MIC or an MS-MPC, you must create a firewall filter that has **sample** configured as action, and apply that to the interface on which you want to monitor the traffic. The flow-collector in Junos Traffic Vision implementations is a device for collecting the flow records. The flow collector is typically deployed outside the network.

NOTE: For more information about configuring firewall filters, see the Junos OS *Firewall Filters Configuration Guide*.

On MS-MIC and MS-MPC, Junos OS supports Junos Traffic Vision Version 9 (v9). Junos Traffic Vision v9 supports sampling of IPv4, IPv6, and MPLS traffic. A services interface card is essential for the v9 implementation, and hence this is often known as PIC-based monitoring.

You can configure the maximum time for which the flow records are stored on the services interface card. The active timeout and inactive timeout values, configured while defining the template, control the export of flow records to the collector. An MS-MIC can store a maximum of 14 million flow records, whereas an MS-MPC can store upto 30 million flows per NPU.

NOTE: In Junos Traffic Vision configurations using the Junos OS extension-provider package, modifying the following statements after flow monitoring has been initiated causes all existing flows to expire:

- At the **[edit forwarding-options sampling instance *instance-name* family (inet |inet6 |mpls) output]** and **[edit forwarding-options sampling family (inet |inet6 |mpls) output]** hierarchy levels:
  - **flow-server *ip-address***
  - **flow-server port *port-number***
  - **flow-server template *template***
- At the **[edit services flow-monitoring version9 template *template-name* mpls-ipv4-template]** and **[edit services flow-monitoring version9 template *template-name* mpls-template]** hierarchy levels:
  - **label-position**

Because these changes can disrupt the ongoing flow monitoring, we recommend that you do not change these values after flow monitoring has been initiated on a device. The changes made to these configuration statements when flow monitoring is going on, apply only to the newly created flows.

Also, note that these changes do not disrupt flow monitoring on devices running Jflow configuration using the Junos OS Layer 2 services package. However, even in the case of Layer 2 service package-based configuration, the changes are applied only to the newly created flows. The existing flows continue to use the initial settings.

NOTE: When Junos Traffic Vision is configured on the MS-MIC and MS-MPC, the next-hop address and outgoing interfaces are incorrectly displayed in the IPv4 and IPv6 flow records when the destination of the sampled flow is reachable through multiple paths.

Configuring Flow Monitoring on MS-MIC

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.

NOTE: You can follow the same procedure and use the same configuration for configuring flow monitoring on MS-MPC.

Enabling the Services Interface Card

```
set interfaces ms-2/0/0 unit 0 family inet
```

Configuring the Template and Timers

```
set services flow-monitoring version9 template template1
set services flow-monitoring version9 template template1 flow-active-timeout 120
set services flow-monitoring version9 template template1 flow-inactive-timeout 60
set services flow-monitoring version9 template template1 ipv4-template
set services flow-monitoring version9 template template1 template-refresh-rate packets 100
set services flow-monitoring version9 template template1 template-refresh-rate seconds 600
set services flow-monitoring version9 template template1 option-refresh-rate packets 100
set services flow-monitoring version9 template template1 option-refresh-rate seconds 600
```

Configuring Service Set Properties

```
set services service-set ss1 jflow-rules sampling
set services service-set ss1 sampling-service service-interface ms-2/0/0.0
```

Configuring Forwarding Options and Flow Server Settings

```
set forwarding-options sampling input rate 10
set forwarding-options sampling input run-length 18
set forwarding-options sampling family inet output flow-server 10.44.4.3 port 1055
set forwarding-options sampling family inet output flow-server 10.44.4.3 version9 template template1
set forwarding-options sampling family inet output interface ms-2/0/0.0 source-address 203.0.113.1
```

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

NOTE: The MS interface must be configured with the family type that the collector will be reachable by. If the collector for the sampling traffic is reachable via IPv4, you must set the family inet under the MS interface even if you are only sampling IPv6 and MPLS traffic, for example.

1. Configure the services interface.

```
[edit interfaces]
user@router1# set interfaces ms-2/0/0 unit 0 family inet
user@router1# set interfaces ms-2/0/0 unit 1 family inet6
user@router1# set interfaces ms-2/0/0 unit 2 family mpls
```

2. Configure the template properties and the export policy timers.

```
[edit services]
user@router1# set flow-monitoring version9 template template1
user@router1# set flow-monitoring version9 template template1 flow-active-timeout 120
user@router1# set flow-monitoring version9 template template1 flow-inactive-timeout 60
user@router1# set flow-monitoring version9 template template1 ipv4-template
user@router1# set flow-monitoring version9 template template1 template-refresh-rate packets 100
user@router1# set flow-monitoring version9 template template1 template-refresh-rate seconds 600
user@router1# set flow-monitoring version9 template template1 option-refresh-rate packets 100
user@router1# set flow-monitoring version9 template template1 option-refresh-rate seconds 600
```

Table 25: Quick Reference to Key Configuration Statements at This Hierarchy Level

Configuration Statement	Description
flow-active-timeout	Configures the interval (in seconds) after which an active flow is exported. Range is 10 through 600 seconds, and the default value is 60 seconds.
flow-inactive-timeout	Configures the interval (in seconds) of inactivity after which a flow is marked inactive. Range is 10 through 600 seconds, and the default value is 60 seconds.
ipv4-template* | *ipv6-template* | *mpls-template* | *mpls-ipv4-template	Specifies the type of traffic for which the template is used for.
template-refresh-rate	Specifies the template refresh rate either as number of packets (range is 1 through 480,000 and the default value is 4800) or in seconds (the range is 10 through 600 and the default is 60). Because the communication between the flow generator and the flow collector is a one-way communication, the flow generator has to regularly send updates about template definitions to the flow collector. The value configured for this statement controls the frequency of such updates.
option-refresh-rate	Specifies the option refresh rate either as number of packets (range is 1 through 480,000 and the default value is 4800) or in seconds (the range is 10 through 600 and the default is 60).

3. Configure service set properties.

```
[edit services]
user@router1# set service-set ss1 jflow-rules sampling
user@router1# set service-set ss1 sampling-service service-interface ms-2/0/0.0
```

Table 26: Quick Reference to Configuration Statements at This Hierarchy Level

Configuration Statement	Description
sampling	Configures the service set to handle sampling/flow monitoring activities.

Table 26: Quick Reference to Configuration Statements at This Hierarchy Level *(continued)*

Configuration Statement	Description
service-interface	Specifies the service interface associated with the service set. The interface configured here should match the interface configured at the **[edit forwarding-options sampling family inet output]**. Also, note that the interface should not be associated with any other service set.

4. Configure forwarding options and flow-server properties.

```
[edit forwarding-options]
user@router1# set sampling input rate 10
user@router1# set sampling input run-length 18
user@router1# set sampling family inet output flow-server 10.44.4.3 port 1055
user@router1# set sampling family inet output flow-server 10.44.4.3 version9 template template1
user@router1# set sampling family inet output interface ms-2/0/0.0 source-address 203.0.113.1
```

NOTE: You can specify the sampling parameters either at the global level (as shown in this example) or at the FPC level by defining a sampling instance. To define a sampling instance, include the **instance** statement at the **[edit forwarding-options sampling]** hierarchy level, and the **sampling-instance** statement at the **[edit chassis fpc *number*]** hierarchy level to associate the sampling instance with an FPC. Under the **[edit forwarding-options sampling instance *instance*]** hierarchy level, you must also include the **input** and **output** configurations explained in this step.

Table 27: Quick Reference to Key Configuration Statements at this Hierarchy Level

Configuration Statement	Description
rate	The ratio of the number of packets to be sampled. For example, if you specify a rate of 10, every tenth packet (1 packet out of 10) is sampled. The range is 1 through 16000000(16M).
run-length	The number of samples following the initial trigger event. This enables you to sample packets following those already being sampled. The range is 0 through 20, and the default is 0.
flow-server	A host system to collect sampled flows using the version 9 format.

Table 27: Quick Reference to Key Configuration Statements at this Hierarchy Level *(continued)*

Configuration Statement	Description
source-address	An IPv4 address to be used as the source address of the exported packet.

Result

From the configuration mode, confirm your configuration by entering the **show chassis fpc 2**, **show interfaces**, and **show forwarding-options** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@router1# show interfaces
ms-2/0/0 {
    unit 0 {
        family inet;
    }
}
```

```
user@router1# show services
flow-monitoring {
    version9 {
        template template1 {
            flow-active-timeout 120;
            flow-inactive-timeout 60;
            template-refresh-rate {
                packets 100;
                seconds 600;
            }
            option-refresh-rate {
                packets 100;
                seconds 600;
            }
            ipv4-template;
        }
    }
}
    service-set ss1 {
        jflow-rules {
            sampling;
        }
        sampling-service {
            service-interface ms-2/0/0.0
        }
```

```
    }
```

```
user@router1# show forwarding-options
sampling {
    input {
        rate 10;
        run-length 18;
    }
    family inet {
        output {
            flow-server 10.44.4.3 {
                port 1055;
                version9 {
                    template {
                        template1;
                    }
                }
            }
            interface ms-2/0/0.0 {
                source-address 203.0.113.1;
            }
        }
    }
}
```

Verification

IN THIS SECTION

Confirm that the configuration is working properly.

Verifying the Junos Traffic Vision Configuration

Purpose

Verify that Junos Traffic Vision is enabled on the router.

Action

From operational mode, enter the **show services accounting status** command.

```
user@router1> show services accounting status
```

```
Service Accounting interface: ms-2/0/0
Export format: 9, Route record count: 2093
IFL to SNMP index count: 35, AS count: 2
Configuration set: Yes, Route record set: Yes, IFL SNMP map set: Yes
```

Meaning

Shows the service interface on which monitoring is configured, and also provides information about the export format used (version 9 in this case).

Viewing the Flow Details

Purpose

View the flow details on the interface configured for flow monitoring.

Action

From operational mode, enter the **show services accounting flow** command.

```
user@router1> show services accounting flow
```

```
Flow information
    Service Accounting interface: ms-2/0/0, Local interface index: 229
    Flow packets: 220693, Flow bytes: 24276230
    Flow packets 10-second rate: 99, Flow bytes 10-second rate: 10998
    Active flows: 10, Total flows: 12
    Flows exported: 199, Flows packets exported: 718
    Flows inactive timed out: 2, Flows active timed out: 199
```

Viewing Details of Errors That Occurred on the Services Interface

Purpose

View details of errors, if any, on the interface that is configured for flow monitoring.

Action

From operational mode, enter the **show services accounting errors** command.

```
user@router1> show services accounting errors
```

```
Error information
    Service Accounting interface: ms-2/0/0
    Service sets dropped: 0, Active timeout failures: 0
    Export packet failures: 0, Flow creation failures: 0
    Memory overload: No
```

RELATED DOCUMENTATION

Multiservices MIC and Multiservices MPC (MS-MIC and MS-MPC) Overview

Example: Configuring Junos VPN Site Secure on MS-MIC and MS-MPC

Configuring Services Interface Redundancy with Flow Monitoring

Active monitoring services configurations on AS, Multiservices PICs, and Multiservices DPCs support redundancy. To configure redundancy, you specify a redundancy services PIC (**rsp**) interface in which the primary AS or Multiservices PIC is active and a secondary PIC is on standby. If the primary PIC fails, the secondary PIC becomes active, and all service processing is transferred to it. If the primary PIC is restored, it remains on standby and does not preempt the secondary PIC; you need to manually restore the services to the primary PIC. To determine which PIC is currently active, issue the **show interfaces redundancy** command.

NOTE: On flow-monitoring configurations, the only service option supported is *warm standby*, in which one backup PIC supports multiple working PICs. Recovery times are not guaranteed, because the configuration must be completely restored on the backup PIC after a failure is detected. However, configuration is preserved and available on the new active PIC.

As with the other services that support warm standby, you can issue the **request interfaces (revert | switchover)** command to switch manually between the primary and secondary flow monitoring interfaces.

For more information, see *Configuring AS or Multiservices PIC Redundancy*. For information on operational mode commands, see the CLI Explorer.

A sample configuration follows.

```
interface {
```

```
    rsp0 {
        redundancy-options {
            primary sp-0/0/0;
            secondary sp-1/3/0;
        }
        unit 0 {
            family inet;
        }
    }
}
interface {
    ge-0/2/0 {
        unit 0 {
            family inet {
                filter {
                    input as_sample;
                }
            }
            address 10.58.255.49/28;
        }
    }
}
forwarding-options {
    sampling {
        instance instance1 { # named instances of sampling parameters
            input {
                rate 1;
                run-length 0;
                max-packets-per-second 65535;
            }
            family inet {
                output {
                    flow-server 10.10.10.2 {
                        port 5000;
                        version 5;
                    }
                    flow-active-timeout 60;
                    interface rsp0 {
                        source-address 10.10.10.1;
                    }
                }
            }
        }
    }
```

```
}
firewall {
    filter as_sample {
        term t1 {
            then {
                sample;
                accept;
            }
        }
    }
}
```

RELATED DOCUMENTATION

- Active Flow Monitoring Overview | 45
- Configuring Flow Monitoring | 3
- Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | 540
- Example: Configuring Active Monitoring on an M, MX or T Series Router's Logical System | 48

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250

Active flow monitoring is implemented on the Packet Forwarding Engine. The Packet Forwarding Engine performs functions such as creating and updating flows, and updating flow records. The flow records are sent out in industry-standard IPFIX or version 9 format. Support for active flow monitoring with IPFIX templates on QFX10002 switches was added in Junos OS Release 17.2R1.

On routers with MS-PICs or MS-DPCs, IPv4 and IPv6 fragments are processed accurately. The flow monitoring application creates two flows for every fragmented flow. The first fragment that has the complete Layer 4 information forms the first flow with 5-tuple data and subsequently, all the fragmented packets related to this flow form another flow with the Layer 4 fields set to zero.

The following considerations apply to the inline flow-monitoring instance configuration:

- Sampling run-length and clip-size are not supported.
- For inline configurations, collectors are not reachable via **fxp0**.
- Inline flow monitoring does not support **cflowd**. Therefore, inline flow monitoring does not support the local dump option, which is available only with cflowd.

- Inline active flow monitoring is not supported when you enable Next Gen Services on an MX Series router.
- The number of collectors that are supported depends on the device:
  - On MX Series routers running Junos OS Release 16.1R4 and later, you can export flow records to four collectors under a family with the same source IP address for Inline-JFlow. The Packet Forwarding Engine (PFE) can export the flow record, flow record template, option data, and option data template packet to all configured collectors. You can configure the multiple collectors at the **[edit forwarding-options sampling instance instance name]** hierarchy level.
  - For inline configurations on all other devices, each family can support only one collector.

Inline active flow monitoring is available in four hierarchies levels:

- **[edit chassis]** —At this level, you associate the sampling instance with the FPC on which the media interface is present (except on the MX80 and MX104—see "Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers" on page 455). If you are configuring sampling of IPv4 flows, IPv6 flows or VPLS flows, you can configure the flow hash table size for each family, as described below.
- **[edit firewall]**—At this level, you configure a firewall filter for the family of traffic to be sampled. You must attach this filter to the interface on which you want to sample the traffic.
- **[edit forwarding-options]**—At this level, you configure a sampling instance and associate the template with the sampling instance. At this level, you also configure the flow-server IP address and port number as well as the flow export rate.
- **[edit services flow-monitoring]** —At this level, you configure the template properties for inline flow monitoring.

Before you configure inline active flow monitoring, you should ensure that you have adequately-sized hash tables for IPv4, IPv6, MPLS, and VPLS flow sampling. These tables can use one to fifteen 256K areas. Starting with Junos OS Release 16.1R1 and 15.1F2, the IPv4 table is assigned a default value of 1024. Prior to Junos OS Release 16.1 and 15.1F2, the IPv4 table is assigned a default value of fifteen 256K areas. The IPv6 table is assigned a default value of 1024, and the VPLS table is assigned a default value of 1024. When anticipated traffic volume requires larger tables, allocate larger tables.

To allocate flow hash tables:

1. Go to the [edit-flow-table-size] hierarchy level for inline services on the FPC that processes the monitored flows.

   ```
   [edit]
   user@host# edit chassis fpc 0 inline-services flow-table-size
   ```

2. Specify the required sizes for the sampling hash tables.

```
[edit chassis fpc 0 inline-services flow-table-size]
user@host# set bridge-flow-table-sizeunits
user@host# set ipv4-flow-table-size units
user@host# set ipv6-flow-table-size units
user@host# set mpls-flow-table-size units
user@host#set vpls-flow-table-size units
```

NOTE: Starting in Junos OS Release 18.2R1, the **bridge-flow-table-size** option is available and the **vpls-flow-table-size** option is deprecated; use the **bridge-flow-table-size** option instead. The **bridge-flow-table-size** option supports both VPLS and bridge records.

NOTE: The total number of units used for IPv4, IPv6, MPLS, and VPLS cannot exceed 15. Also, starting in Junos OS Release 16.1R1 and 15.1F2, changing the flow hash table size does *not* automatically reboot the FPC (for earlier releases changing the flow hash table size triggers the FPC to reboot).

To configure inline active flow monitoring on MX Series routers (except for MX80 and MX104 routers), EX Series switches, and T4000 routers with Type 5 FPC:

1. Enable inline active flow monitoring and specify the source address for the traffic.

```
[edit forwarding-options sampling instance instance-name family (bridge | inet | inet6 | mpls | vpls ) output]
user@host# set inline-jflow source address address
```

2. Specify the template to use with the sampling instance.

```
[edit forwarding-options sampling instance instance-name family (bridge | inet | inet6 | mpls | vpls ) output
    flow-server hostname]
user@host# set (version9 | version-ipfix) template template-name
```

3. Configure a template to specify output properties.

```
[edit services flow-monitoring]
user@host# set (version-ipfix | version9) template template-name
```

4. (Optional) Configure the interval after which an active flow is exported.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-active-timeout seconds
```

5. (Optional) Configure the interval of activity that marks a flow as inactive.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-inactive-timeout seconds
```

6. (Optional) Configure the template refresh rate in either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set template-refresh-rate packets packets seconds seconds
```

7. (Optional) Configure the refresh rate in either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set option-refresh-rate packets packets seconds seconds
```

8. Specify the type of record that the template is used for.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set (bridge-templateipv4-template |ipv6-template | mpls-ipv4-template | mpls-template |
   peer-as-billing-template | vpls-template)
```

   The **vpls-template** option is only for IPFIX templates.

   Starting in Junos OS Release 18.2R1, the **bridge-template** option is available and the **vpls-template** option is deprecated; use the **bridge-template** option instead. The **bridge-template** option supports both VPLS and bridge records and is for both IPFIX and version9 templates.

   Starting in Junos OS Release 18.4R1, the **MPLS-ipv4-template** option is deprecated for inline flow monitoring. To configure MPLS records starting in Junos OS Release 18.4R1, use the **mpls-template** option and the **tunnel-observation** option. This is described in step 9.

9. Starting in Junos OS Release 18.4R1 for the MX Series, if you are configuring any type of MPLS flow records, perform the following:

   a. Specify the MPLS template.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
```

```
user@host# set mpls-template
```

b. Configure the type of MPLS flow records to create.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set tunnel-observation [ipv4 | ipv6]
```

The **tunnel-observation** values enable the creation of the following types of flow records:

- **ipv4**—MPLS-IPv4 flows
- **ipv6**—MPLS-IPv6 flows

You can configure multiple values for **tunnel-observation**.

For an MPLS traffic type that does *not* match any of the **tunnel-observation** values, plain MPLS flow records are created. For example, if you only configure **ipv4**, then MPLS-IPv6 traffic results in plain MPLS flow records.

If you do not configure **tunnel-observation**, plain MPLS flow records are created.

c. If you are running inline flow monitoring on a Lookup (LU) card, enable sideband mode to create MPLS-IPv6 flow records.

```
[edit chassis fpc slot-number inline-services]
user@host# set use-extended-flow-memory
```

If you are running inline flow monitoring on an LU card and do not enable sideband mode, then MPLS-IPv6 traffic results in plain MPLS flow records.

10. (Optional) Include the flow direction value in the template.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-key flow-direction
```

The reported data field contains 0x00 (ingress) or 0x01 (egress). If you do not include the **flow-key flow-direction** statement, the flow direction data field contains the invalid value 0xFF.

11. (Optional) Include VLAN IDs in both the ingress and egress directions in the flow key.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-key vlan-id
```

This statement is not required for ingress and egress VLAN ID reporting on interfaces.

12. Associate the sampling instance with the FPC on which you want to implement inline active flow monitoring.

    For MX240, MX480, MX960, MX2010, MX2020, use the following command:

    ```
    [edit ]
    user@host# set chassis fpc fpc-number sampling-instance instance-name.
    ```

    a. Confirm the configuration by running the following show command:

    ```
    user@host# show chassis
    ```

    ```
    fpc 0 {
            sampling-instance sample-ins1;
        }
    ```

    For MX5, MX10, MX40, and MX80, use the following command:

    ```
    [edit ]
    user@host# set chassis tfeb slot 0 sampling-instance instance-name.
    ```

    a. Confirm the configuration by running the following show command:

    ```
    user@host# show chassis
    ```

    ```
    tfeb {
        slot 0 {
            sampling-instance sample-ins1;
        }
    }
    ```

    For MX104, use the following command:

    ```
    [edit ]
    user@host# set chassis afeb slot 0 sampling-instance instance-name.
    ```

    a. Confirm the configuration by running the following show command:

    ```
    user@host# show chassis
    ```

```
afeb {
    slot 0 {
        sampling-instance sample-ins1;
    }
}
```

This example shows the sampling configuration for an instance that supports inline active flow monitoring on **family inet**:

```
[edit]
user@host> show forwarding-options
sampling {
   instance {
      sample-ins1 {
         input {
            rate 1;
         }
         family inet {
            output {
               flow-server 192.0.2.2 {
                  port 2055;
                  version-ipfix {
                     template {
                        ipv4;
                     }
                  }
               }
               inline-jflow {
                  source-address 10.11.12.13;
               }
            }
         }
      }
   }
}
```

Here is the output format configuration:

```
[edit]
user@host> show services flow-monitoring
services {
   flow-monitoring {
      version-ipfix {
```

```
        template ipv4 {
          flow-active-timeout 60;
          flow-inactive-timeout 60;
          ipv4-template;
          template-refresh-rate {
            packets 1000;
            seconds 10;
          }
          option-refresh-rate {
            packets 1000;
            seconds 10;
          }
        }
      }
    }
}
```

The following example shows the output format configuration for chassis **fpc** 0:

```
[edit]
user@host> show services flow-monitoring
sampling-instance instance-1; {
   inline-services {
      flow-table-size {
         ipv4-flow-table-size 8;
         ipv6-flow-table-size 7;
      }
   }
}
```

Release History Table

Release	Description
19.3R2	Inline active flow monitoring is not supported when you enable Next Gen Services on an MX Series router.
18.4R1	Starting in Junos OS Release 18.4R1, the **MPLS-ipv4-template** option is deprecated for inline flow monitoring. To configure MPLS records starting in Junos OS Release 18.4R1, use the **mpls-template** option and the **tunnel-observation** option.
18.2R1	Starting in Junos OS Release 18.2R1, the **bridge-flow-table-size** option is available and the **vpls-flow-table-size** option is deprecated; use the **bridge-flow-table-size** option instead.
18.2R1	Starting in Junos OS Release 18.2R1, the **bridge-template** option is available and the **vpls-template** option is deprecated; use the **bridge-template** option instead.
16.1R4	On MX Series routers running Junos OS Release 16.1R4 and later, you can export flow records to four collectors under a family with the same source IP address for Inline-JFlow.
16.1R1	Starting with Junos OS Release 16.1R1 and 15.1F2, the IPv4 table is assigned a default value of 1024.
16.1R1	Also, starting in Junos OS Release 16.1R1 and 15.1F2, changing the flow hash table size does *not* automatically reboot the FPC

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers | **455**

Example: Configuring Inline Active Flow Monitoring on MX Series and T4000 Routers | **479**

inline-jflow | **991**

Configuring Flow Offloading on MX Series Routers

The Junos OS enables you to configure flow offloading for PICS on MX Series routers using Modular Port Concentrator (MPCs) with Modular Interface Cards (MICs). Flows are offloaded to Fast Update Filters (FUFs) on the Packet Forwarding Engine. Offloading produces the greatest benefits when applied to long-lasting or high-bandwidth flows.

The maximum number of active offloads is 200,000 per PIC. When offloaded flows are deleted, more flows can be offloaded.

To configure flow offloading:

- At the **[edit interfaces *interface-name* services-options]** hierarchy level, enter the **trio-flow-offload minimum-bytes *minimum-bytes*** statement.

```
user@host# edit services interface-name
[edit services interface-name services-options]
user@host# set trio-flow-offload minimum-bytes minimum- bytes
```

In the following example, flows are offloaded when they consist of no less than 1024 bytes:

```
user@host# edit services ms-0/1/0
[edit services ms-0/1/0 services-options]
user@host# set trio-flow-offload minimum-bytes 1024
```

RELATED DOCUMENTATION

trio-flow-offload | 1205

Configuring Active Flow Monitoring on PTX Series Packet Transport Routers

You can use flow monitoring to help with network administration. Active flow monitoring on PTX Series routers allows you to collect sampled packets, then the router does GRE encapsulation of the packets and sends them to a remote server for flow processing. The GRE encapsulation includes an interface index and GRE key field. The GRE encapsulation removes MPLS tags. You configure one or more port-mirroring instances to define which traffic to sample and configure a server to receive the GRE encapsulated packets. You configure a firewall filter on interfaces where you want to capture flows. You can configure as many as 48 port-mirroring instances.

To configure the router to do GRE encapsulation of sampled packets and send them to a remote server for flow processing:

1. Configure one or more server profiles that specify a host where GRE encapsulated sampled packets are sent, and optionally, a source address to include in the header of each sampled packet.

   a. Specify a name for each server profile and an IP address of the host where sampled packets are sent:

   ```
   [edit services hosted-services]
   user@host# set server-profile server-profile-name server-address ipv4-address
   ```

   b. (Optional) For each server profile, specify a source address to include in the header of each sampled packet:

   ```
   [edit services hosted-services server-profile server-profile-name]
   user@host# set client-address ipv4-address
   ```

   NOTE: The default client address is 0.0.0.0. You must specify an IPv4 address as the client address. You can also specify the loopback address or management interface address as the client address.

2. Configure one or more port-mirroring instances.

   a. Specify a name for each port-mirroring instance:

   ```
   [edit forwarding-options port-mirroring]
   user@host# set instance instance-name
   ```

NOTE: You can configure a maximum of 48 port-mirroring instances.

b. Specify a protocol family for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name]
user@host# set family (inet | inet6 )
```

3. To set the ratio of the number of packets to sample, specify a value from 1 through 65,535 for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name input]
user@host# set rate number
```

NOTE: You must specify a value for the **rate** statement. The default value is zero, which effectively disables sampling. If, for example, you specify a rate value of 4, every fourth packet (1 packet out of 4) is sampled.

4. (Optional) Specify the number of samples to collect after the initial trigger event for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name input]
user@host# set run-length number
```

NOTE: The default value is zero. You can specify a number up to 20.

5. To designate a host where sampled traffic is sent, specify the name of server profile configured at the **[edit services hosted-services]** hierarchy level for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name family ( inet | inet6) output]
user@host# set server-profile server-profile-name
```

6. Configure one or more firewall filters.

   a. For each firewall filter, specify a protocol family, filter name, and match conditions:

      ```
      [edit firewall]
      user@host# set filter family (inet | inet6) filter filter-name term term-name from match-condtions
      ```

   b. For each firewall filter you configure, specify the name of a port-mirroring instance you configured at the **[edit forwarding-options]** hierarchy level as a nonterminating action so that the traffic that matches that instance is sampled:

      ```
      [edit firewall family (inet | inet6) filter filter-name term term-name]
      user@host# set then port-mirroring instance instance-name
      ```

7. Apply each firewall filter to an interface to evaluate incoming traffic:

   ```
   [edit interfaces interface-name unit logical-unit-number]
   user@host# set family (inet | inet6) filter input firewall-filter-name
   ```

   NOTE: Active flow monitoring is supported only on incoming traffic. You cannot apply firewall filters to evaluate outgoing traffic.

8. Configure the remote server, where GRE encapsulated packets are sent, to perform flow processing.

RELATED DOCUMENTATION

- *Configuring Port Mirroring*
- *hosted-services*
- *port-mirroring*
- *server-profile (Active Flow Monitoring)*
- *Firewall Filter Nonterminating Actions*

Configuring Actively Monitored Interfaces on M, MX and T Series Routers

Configure the input interfaces and apply the firewall filter that you defined earlier. Unlike passive flow monitoring, the input interfaces for active flow monitoring are not restricted, so you can select most standard network interfaces (such as ATM1 or Ethernet-based interfaces) as the input.

If you configure active flow monitoring with sampling, you can configure an interface filter in place of a firewall filter with the **sampling** statement at the **[edit interfaces *interface-name-fpc/pic/port* unit *unit-number* family inet]** hierarchy level.

```
[edit]
interfaces {
   so-2/2/0 {
      unit 0 {
         family inet {
            filter {
               input active_filter;
            }
            address 10.36.11.2/32 {
               destination 10.36.11.1;
            }
            sampling {
               (input | output | [input output]);
            }
         }
      }
   }
}
```

Collecting Flow Records

Traffic flows can be exported in flow monitoring version 5, 8, and 9 formats for active flow monitoring. The default export format for flow monitoring records is version 5. To change the export format to flow monitoring version 8, include the **version 8** statement at either the [**edit forwarding-options accounting *name* output flow-server *flow-server-address***] or the [**edit forwarding-options sampling output flow-server *flow-server-address***] hierarchy level. To change the export format to flow monitoring version 9, include the **version9 template *template-name*** statement at the [**edit forwarding-options sampling output flow-server *flow-server-address***] hierarchy level. For more information on flow record formats, see “Flow Monitoring Output Formats” on page 9.

To capture flow data generated by the Monitoring Services PIC, Adaptive Services PIC, or MultiServices PIC and export it to a flow server, you can use one of the following active flow monitoring methods:

- Configuring M, MX and T Series Routers for Discard Accounting with a Sampling Group on page 78
- Configuring M, MX and T Series Routers for Discard Accounting with an Accounting Group on page 77
- Configuring M, MX and T Series Routers for Discard Accounting with a Template on page 79
- Replicating M, MX and T Series Routing Engine-Based Sampling to Multiple Flow Servers on page 83
- Replicating Version 9 Flow Aggregation From M, MX and T Series Routers to Multiple Flow Servers on page 84
- Configuring Routing Engine-Based Sampling on M, MX and T Series Routers for Export to Multiple Flow Servers on page 85
- Configuring an Aggregate Export Timer on M, MX and T Series Routers for Version 8 Records on page 110

Configuring M, MX and T Series Routers for Discard Accounting with an Accounting Group

To perform discard accounting on specified traffic, you can collect flow records with the **accounting** statement at the **[edit forwarding-options]** hierarchy level. Like sampling, your topology must be simple (for example, one input interface and one export interface).

Again, you can collect flow records by specifying input and output interfaces. You can configure the input interface to perform discard accounting by applying a firewall filter that contains the **then discard accounting** statement. This match condition directs the filtered traffic to be converted into flow records and exported for analysis by the monitoring services or adaptive services interface. The original packets are then sent to the discard process. For the output, remember to specify the IP address and port of your flow server and the services interface you plan to use for processing flow records.

You must configure a source address, but the **engine-id** and **engine-type** output interface statements are added automatically. You can override these values manually to track different flows with a single flow collector. SNMP input and output interface index information is captured in flow records by default when you configure discard accounting.

```
[edit]
forwarding-options {
    accounting counter1 {
        output {
            flow-inactive-timeout 65;
            flow-active-timeout 65;
```

```
        flow-server 10.60.2.1 {
            port 2055;
            version 8;
            aggregation {
                protocol-port;
                source-destination-prefix;
            }
        }
        interface sp-2/0/0 {
            engine-id 1;
            engine-type 11;
            source-address 10.60.2.2;
        }
    }
  }
}
```

Configuring M, MX and T Series Routers for Discard Accounting with a Sampling Group

If your needs for active flow monitoring are simple, you can collect flow records with a sampling group. Sampling does not require you to configure a monitoring group (as required in passive flow monitoring) because you can configure flow server information in the **sampling** hierarchy. When you wish to sample traffic, include the **sampling** statement at the **[edit forwarding-options]** hierarchy level.

The typical sampling configuration has one input interface and one export interface. The input interface is activated by the **then sample** statement in a firewall filter term. This match condition directs traffic to the sampling process. Alternatively, you can use an interface-based filter in place of a firewall filter if you include the **sampling** statement at the **[edit interfaces *interface-name-fpc/pic/port* unit *unit-number* family inet]** hierarchy level.

There are two types of sampling available: PIC-based sampling and Routing Engine-based sampling. PIC-based sampling occurs when a monitoring services or adaptive services interface is the target for the output of the sampling process. To enable PIC-based sampling, include the **interface** statement at the **[edit forwarding-options sampling output]** hierarchy level and specify a monitoring services or adaptive services interface as the output interface. If an output interface is not specified in the sampling configuration, sampling is performed by the Routing Engine.

To specify a flow server in a sampling configuration, include the **flow-server** statement at the **[edit forwarding-options sampling output]** hierarchy level. You must specify the IP address, port number, and flow monitoring version of the destination flow server. Routing Engine-based sampling supports flow

aggregation of up to eight flow servers (version 5 servers and version 8 only) at a time. The export packets are replicated to all flow servers configured to receive them. In contrast, PIC-based sampling allows you to specify just one version 5 flow server and one version 8 server simultaneously. Flow servers operating simultaneously must have different IP addresses.

As part of the output interface statements, you must configure a source address. In contrast, the interface-level statements of **engine-id** and **engine-type** are both added automatically. However, you can override these values with manually configured statements to track different flows with a single flow collector, as needed. When you configure sampling, SNMP input and output interface index information is captured in flow records by default.

```
[edit]
forwarding-options {
    sampling {
        input {
            rate 1;
        }
        family inet {
            output {
                flow-inactive-timeout 15;
                flow-server 10.60.2.1 {
                    port 2055;
                    version 5;
                }
                interface sp-2/0/0 {
                    engine-id 5;
                    engine-type 55;
                    source-address 10.60.2.2;
                }
            }
        }
    }
}
```

Configuring M, MX and T Series Routers for Discard Accounting with a Template

Flow monitoring version 9, which is based on RFC 3954, provides a way to organize flow data into templates. Version 9 also provides a way to actively monitor IPv4, IPv6, MPLS, and peer AS billing traffic. Version 9 is not supported on the AS-I PIC.

To activate templates in flow monitoring, you must configure a template and include that template in the version 9 flow monitoring configuration. Version 9 does not work in conjunction with versions 5 and 8.

To configure a version 9 template, include the **template *template-name*** statement at the [**edit services flow-monitoring version9**] hierarchy level. The Junos OS supports five different templates: **ipv4-template**, **ipv6–template**, **mpls-template**, **mpls-ipv4-template**, and **peer-as-billing-template**. To view the fields selected in each of these templates, see "Flow Monitoring Version 9 Format Output Fields" on page 21.

```
[edit]
services flow-monitoring {
   version9 { # Specifies flow monitoring version 9.
      template mpls { # Specifies template you are configuring.
         template-refresh-rate {
            packets 6000; # The default is 4800 packets and the range is 1–480000
            # packets.
         seconds 90; # The default is 60 seconds and the range is 1–600 seconds.
            option--refresh-rate {
            packets 3000; # The default is 4800 packets and the range is 1–480000
               # packets.
            seconds 30; # The default is 60 seconds and the range is 1–600.
            flow-active-timeout 60; # The default is 60 seconds and the range is
               # 10–600.
            flow-inactive-timeout 30; # The default is 60 seconds and the range 10–600.
            template-refresh-rate seconds 10; # The default is 60 seconds and the
               # range is 10–600
            option-refresh-rate seconds 10; # The default is 60 seconds and the range
               # is 10–600 seconds.
            mpls-template {
               label-positions [1 | 2 | 3]; # Specifies label position for the MPLS template.
            }
         }
      }
   }
}
```

You can export to multiple templates at a time to a maximum of eight flow servers for AS PICs and one flow server for all other PICs. To assign a template to a flow output, include the **template *template-name*** statement at the [**edit forwarding options sampling output flow-server version9**] hierarchy level:

```
[edit]
forwarding-options {
   sampling {
      input {
         family mpls {
```

```
            rate 1;
            run-length 1;
          }
        }
        output {
          flow-server 10.60.2.1 { # The IP address and port of the flow server.
            port 2055;
            source-address 192.0.2.1;
            version9 { # Records are sent to the flow server using version 9 format.
              template { # Indicates a template will organize records.
                mpls; # Records are sent to the MPLS template.
              }
            }
          }
        }
      }
    }
```

Defining a Firewall Filter on M, MX and T Series Routers to Select Traffic for Active Flow Monitoring

The first step in active flow monitoring is to configure the match conditions for acceptable traffic or quarantined traffic. Common match actions for active flow monitoring include **sample**, **discard accounting**, **port-mirror**, and **accept**. To configure, include the desired action statements and a counter as part of the **then** statement in a firewall filter and apply the filter to an interface.

In sampling, the router reviews a portion of the traffic and sends reports about this sample to the flow monitoring server. Discard accounting traffic is counted and monitored, but not forwarded out of the router. Port-mirrored traffic is copied and sent to another interface. Accepted traffic is forwarded to the intended destination.

Most of these match combinations are valid. However, you can either port-mirror or sample with the same traffic at the same time, but not perform more than one action simultaneously on the same packets.

```
[edit]
firewall {
  family inet {
    filter active_filter {
      term quarantined_traffic {
        from {
```

```
            source-address {
               10.36.1.2/32;
            }
         }
         then {
            count quarantined-counter;
            sample;
            discard accounting;
         }
      }
      term copy_and_forward_the_rest {
         then {
            port-mirror;
            accept;
         }
      }
   }
}
```

Processing IPv4 traffic on an M, MX or T Series Router Using Monitoring services, Adaptive services or Multiservices Interfaces

You configure the monitoring services, adaptive services, or multiservices interfaces with the **family inet** statement so they can process IPv4 traffic. However, you must remember that a monitoring services interface uses an **mo-** prefix and adaptive services and multiservices interfaces use an **sp-** prefix.

```
[edit]
interfaces {
   sp-2/0/0 {
      unit 0 {
         family inet {
            address 10.36.100.1/32 {
               destination 10.36.100.2;
            }
         }
      }
   }
}
```

Active flow monitoring records leave the router through an export interface to reach the flow monitoring server.

```
[edit]
interfaces {
   fe-1/0/0 {
      unit 0 {
         family inet {
            address 10.60.2.2/30;
         }
      }
   }
}
```

Replicating M, MX and T Series Routing Engine-Based Sampling to Multiple Flow Servers

Routing Engine-based sampling supports up to eight flow servers for both flow monitoring version 5 and version 8 configurations. The total number of flow servers is limited to eight, regardless of how many are configured for version 5 or version 8.

When you configure version 5 or version 8 sampling, the export packets are replicated to all flow servers configured to receive them. If two flow servers are configured to receive version 5 records, both flow servers will receive records for a specified flow.

NOTE: With Routing-Engine-based sampling, if multiple flow servers are configured with version 8 export format, all of them must use the same aggregation type (for example, all flow servers receiving version 8 export could be configured for source-destination aggregation type).

The following configuration example allows replication of export packets to two flow servers.

```
[edit]
forwarding-options {
   sampling {
      input {
         rate 1;
         }
      }
```

```
        output {
            flow-server 10.10.3.2 {
                port 2055;
                version 5;
                source-address 192.168.164.119;
            }
            flow-server 172.17.20.62 {
                port 2055;
                version 5;
                source-address 192.168.164.119;
            }
        }
    }
}
```

Replicating Version 9 Flow Aggregation From M, MX and T Series Routers to Multiple Flow Servers

With this feature, you can configure up to eight flow servers to receive packets for a version 9 flow monitoring template. Once a flow server is configured to receive this data, it will also receive the following periodic version 9 flow monitoring updates:

- Options data
- Template definition

With Routing Engine-based sampling, if multiple collectors are configured with version 8 export format, all of them must use the same aggregation-type.

The option and template definition refresh period is configured on a per-template basis at the [**edit services flow-monitoring**] hierarchy level.

The following configuration example allows replication of version 9 export packets to two flow servers.

```
forwarding-options {
    sampling {
        input {
            family inet {
                rate 1;
            }
        }
        output {
```

```
            flow-server 10.10.3.2 {
                port 2055;
                version9 {
                    template {
                        ipv4;
                    }
                }
            }
            flow-server 172.17.20.62 {
                port 2055;
                version9 {
                    template {
                        ipv4;
                    }
                }
            }
            flow-inactive-timeout 30;
            flow-active-timeout 60;
            interface sp-4/0/0 {
                source-address 10.10.3.4;
            }
        }
    }
}
```

RELATED DOCUMENTATION

- Active Flow Monitoring Overview | 45
- Active Flow Monitoring Overview | 44
- Active Flow Monitoring Applications | 39
- Replicating M, MX and T Series Routing Engine-Based Sampling to Multiple Flow Servers | 83

Configuring Routing Engine-Based Sampling on M, MX and T Series Routers for Export to Multiple Flow Servers

Routing Engine-based sampling supports up to eight flow servers for both version 5 and version 8 configurations. The total number of collectors is limited to eight, regardless of how many are configured for version 5 or version 8. When you configure sampling, the export packets are replicated to all collectors

configured to receive them. If two collectors are configured to receive version 5 records, both collectors will receive records for a specified flow.

The following configuration example allows replication of export packets to two collectors.

```
forwarding-options {
   sampling {
      input {
         family inet {
            rate 1;
         }
      }
      output {
         cflowd 10.10.3.2 {
            port 2055;
            version 5;
            source-address 192.168.164.119;
         }
         cflowd 172.17.20.62 {
            port 2055;
            version 5;
            source-address 192.168.164.119;
         }
      }
   }
}
```

Example: Copying Traffic to a PIC While an M, MX or T Series Router Forwards the Packet to the Original Destination

IN THIS SECTION

Traffic sampling enables you to copy traffic to a Physical Interface Card (PIC) while the router forwards the packet to its original destination. This example describes how to configure a router to perform sampling on the Routing Engine using the **sampled** process. For this method, you configure a filter (input or output) with a matching term that contains the **then sample** statement. In addition, for VPN routing and forwarding (VRF) Routing Engine-based sampling, you configure a VRF routing instance that maps to an interface. Each VRF instance corresponds with a forwarding table. Routes on the interface go into the corresponding forwarding table.

For VRF Routing Engine-based sampling, the kernel queries the correct VRF route table based on the ingress interface index for the received packet. For interfaces configured in VRF, the sampled packets contain the correct input and output interface SNMP index, the source and destination AS numbers, and the source and destination mask.

> **NOTE:** With Junos OS Release 10.1, VRF Routing Engine-based sampling is performed only on IPv4 traffic. You cannot use Routing Engine-based sampling on IPv6 traffic or on MPLS label-switched paths.

This example describes how to configure and verify VRF Routing Engine-based sampling on one router in a four-router topology.

Requirements

This example uses the following hardware and software components:

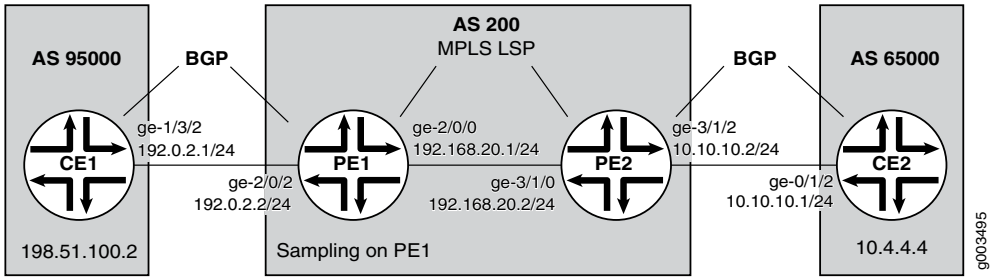
- Junos OS Release 10.1 or later
- M Series, MX Series, or T Series router

Before you configure VRF Routing Engine-Based sampling on your router, be sure you have an active connection between the routers on which you configure sampling. In addition, you need to have an understanding of VRF to configure the interfaces and routing instances that form the basis of the sampling configuration; and an understanding of the BGP, MPLS, and OSPF protocols to configure the other routers in the network to bring up the sampling configuration.

Overview and Topology

The scenario in this example illustrates VRF Routing Engine-based sampling configured on the PE1 router in a four-router network. The CE routers use BGP as the routing protocol to communicate with the PE routers. MPLS LSPs pass traffic between the PE routers. Packets from the CE1 router are sampled on the PE1 router. Regular traffic is forwarded to the original destination (the CE2 router).

Figure 16: Routing Engine-Based Sampling Network Topology



Configuration

IN THIS SECTION

In this configuration example, the VRF Routing Engine-based sampling is configured on the PE1 router that samples the traffic that goes through the interface and routes configured in the VRF. The configurations on the other three routers are included to show the sampling configuration on the PE1 router working in the context of a network.

To configure VRF Routing Engine-based sampling for the network example, perform these tasks:

Configuring the CE1 Router

Step-by-Step Procedure

In this step, you configure interfaces, routing options, protocols, and policy options for the CE1 router. To configure the CE1 router:

1. Configure one interface with two IP addresses. One address is for traffic to the PE1 router; the other address is to check that traffic is flowing to the CE2 router:

   ```
   [edit interfaces]
   user@router-ce1# set ge-1/3/2 unit 0 family inet address 192.0.2.1/24
   user@router-ce1# set ge-1/3/2 unit 0 family inet address 198.51.100.2/8
   ```

2. Configure the autonomous system to establish a connection between BGP peers:

   ```
   [edit routing-options]
   user@router-ce1# set autonomous-system 95000
   ```

3. Configure BGP as the routing protocol between the CE router and the PE router:

   ```
   [edit protocols]
   user@router-ce1# set bgp group to_r1 type external
   user@router-ce1# set bgp group to_r1 export my_lo0_addr
   user@router-ce1# set bgp group to_r1 peer-as 200
   user@router-ce1# set bgp group to_r1 neighbor 192.0.2.2
   ```

4. Configure the policies that ensure that the CE routers exchange routing information. In this example, Router CE1 exchanges routing information with Router CE2:

   ```
   [edit policy-options]
   user@router-ce1# set policy-statement my_lo0_addr term one from protocol direct
   user@router-ce1# set policy-statement my_lo0_addr term one from route-filter 10.255.15.32/32
   exact
   user@router-ce1# set policy-statement my_lo0_addr term one then accept
   user@router-ce1# set policy-statement my_lo0_addr term four from protocol direct
   user@router-ce1# set policy-statement my_lo0_addr term four from route-filter 203.0.113.0/8
   exact
   user@router-ce1# set policy-statement my_lo0_addr term four then accept
   ```

Results

The output below shows the configuration of the CE1 router:

```
[edit]
user@router-ce1# show
```

```
[...Output Truncated...]
interfaces {
    ge-1/3/2 {
        unit 0 {
            family inet {
                address 192.0.2.1/24;
                address 198.51.100.2/8;
            }
        }
    }
}
routing-options {
    autonomous-system 95000;
}
protocols {
    bgp {
        group to_r1 {
            type external;
            export my_lo0_addr;
            peer-as 200;
            neighbor 192.0.2.2;
        }
    }
}
policy-options {
    policy-statement my_lo0_addr {
        term one {
            from {
                protocol direct;
                route-filter 10.255.15.32/32 exact;
            }
            then accept;
        }
        term four {
            from {
                protocol direct;
                route-filter 203.0.113.0/8 exact;
            }
            then accept;
        }
    }
}
```

Configuring the PE1 Router

Step-by-Step Procedure

In this step, you configure a filter with a matching term that contains the **then sample** statement and apply the filter to the ingress interface. You also configure a VRF routing instance with import and export policies. In addition, you configure interfaces, forwarding options, routing options, protocols, and policy options for the PE1 router. To configure the PE1 router:

1. Create the **fw** firewall filter that is applied to the logical interface being sampled:

```
[edit firewall]
user@router-pe1# set family inet filter fw term 1 from protocol tcp
user@router-pe1# set family inet filter fw term 1 from port bgp
user@router-pe1# set family inet filter fw term 1 then accept
user@router-pe1# set family inet filter fw term 2 then sample
```

2. Configure two interfaces, one interface that connects to the CE1 router (**ge-2/0/2**), and another that connects to the PE2 router (**ge-2/0/0**):

```
[edit interfaces]
user@router-pe1# set ge-2/0/2 unit 0 family inet address 192.0.2.2/24
user@router-pe1# set ge-2/0/0 unit 0 family inet address 192.168.20.1/24
user@router-pe1# set ge-2/0/0 unit 0 family mpls
```

3. Enable MPLS on the interface that connects to the PE2 router (**ge-2/0/0**):

```
[edit interfaces]
user@router-pe1# set ge-2/0/0 unit 0 family mpls
```

4. On the interface that connects to the CE1 router, apply the **fw** filter that was configured in the firewall configuration:

```
[edit interfaces]
user@router-pe1# set ge-2/0/2 unit 0 family inet filter input fw
user@router-pe1# set ge-2/0/2 unit 0 family inet filter output fw
```

5. Configure the management (**fxp0**) and loopback (**lo0**) interfaces:

```
[edit interfaces]
user@router-pe1# set fxp0 unit 0 family inet address 192.168.69.153/21
user@router-pe1# set lo0 unit 0 family inet address 127.0.0.1/32
```

6. Configure the **sampled** log file in the **/var/log** directory to record traffic sampling:

```
[edit forwarding-options]
```

```
user@router-pe1# set sampling traceoptions file sampled
user@router-pe1# set sampling traceoptions file world-readable
user@router-pe1# set sampling traceoptions flag all
```

7. Specify the sampling rate and threshold value for traffic sampling:

```
[edit forwarding-options]
user@router-pe1# set sampling input rate 1
user@router-pe1# set sampling input run-length 0
user@router-pe1# set sampling input max-packets-per-second 20000
```

8. Specify active and inactive flow periods, and the router (**198.51.100.2**) that sends out the monitored information:

```
[edit forwarding-options]
user@router-pe1# set sampling family inet output flow-active-timeout 60
user@router-pe1# set sampling family inet output flow-inactive-timeout 60
user@router-pe1# set sampling family inet output flow-server 198.51.100.2 port 2055
user@router-pe1# set sampling family inet output flow-server 198.51.100.2 local-dump
user@router-pe1# set sampling family inet output flow-server 198.51.100.2 version 500
```

9. Configure the autonomous system to establish a connection between BGP peers:

```
[edit routing-options]
user@router-pe1# set autonomous-system 200
```

10. Configure RSVP to support MPLS label-switched paths (LSPs) between the PE routers:

```
[edit protocols]
user@router-pe1# set rsvp interface all
user@router-pe1# set rsvp interface fxp0.0 disable
```

11. Configure an MPLS LSP from the PE1 router to the PE2 router:

```
[edit protocols]
user@router-pe1# set mpls label-switched-path R1toR2 from 192.168.20.1
user@router-pe1# set mpls label-switched-path R1toR2 to 192.168.20.2
user@router-pe1# set mpls interface all
user@router-pe1# set mpls interface fxp0.0 disable
```

12. Configure an internal BGP group for the PE routers. Include the **family inet-vpn unicast** statement to enable BGP to carry network layer reachability information (NLRI) parameters and for BGP peers to only carry unicast routes for forwarding:

```
[edit protocols]
user@router-pe1# set bgp group to_r2 type internal
user@router-pe1# set bgp group to_r2 local-address 192.168.20.1
user@router-pe1# set bgp group to_r2 neighbor 192.168.20.2 family inet-vpn unicast
```

13. Configure OSPF as the interior gateway protocol (IGP) and to compute the MPLS LSPs:

```
user@router-pe1# set ospf traffic-engineering
user@router-pe1# set ospf area 0.0.0.0 interface all
user@router-pe1# set ospf area 0.0.0.0 interface fxp0.0 disable
```

14. Create the extended community that is applied in the policy options configuration:

```
[edit policy-options]
user@router-pe1# set community vpna-comm members target:200:100
```

15. Define the **vpna-export** routing policy that is applied in the **vrf-export** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe1# set policy-statement vpna-export term one from protocol bgp
user@router-pe1# set policy-statement vpna-export term one from protocol direct
user@router-pe1# set policy-statement vpna-export term one then community add vpna-comm
user@router-pe1# set policy-statement vpna-export term one then accept
user@router-pe1# set policy-statement vpna-export term two then reject
```

16. Define the **vpna-import** routing policy that is applied in the **vrf-import** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe1# set policy-statement vpna-import term one from protocol bgp
user@router-pe1# set policy-statement vpna-import term one from community vpna-comm
user@router-pe1# set policy-statement vpna-import term one then accept
user@router-pe1# set policy-statement vpna-import term two then reject
```

17. Configure a VRF routing instance so that routes received from the provider edge-provider edge (PE-PE) session can be imported into any of the instance's VRF secondary routing tables:

```
[edit routing-instances]
user@router-pe1#set vrf1 instance-type vrf set vrf1 interface ge-2/0/2.0
user@router-pe1# set vrf1 route-distinguisher 10.255.15.51:1
user@router-pe1# set vrf1 vrf-import vpna-import
user@router-pe1# set vrf1 vrf-export vpna-export
```

```
user@router-pe1# set vrf1 protocols bgp group customer type external
user@router-pe1# set vrf1 protocols bgp group customer peer-as 95000
user@router-pe1# set vrf1 protocols bgp group customer as-override
user@router-pe1# set vrf1 protocols bgp group customer neighbor 192.168.30.1
user@router-pe1# set vrf1 protocols bgp group customer neighbor 192.0.2.1
```

Results

Check the results of the configuration for the PE1 router:

```
user@router-pe1> show configuration
[...Output Truncated...]
}
interfaces {
    ge-2/0/0 {
        unit 0 {
            family inet {
                address 192.168.20.1/24;
            }
            family mpls;
        }
    }
    ge-2/0/2 {
        unit 0 {
            family inet {
                filter {
                    input fw;
                    output fw;
                }
                address 192.0.2.2/24;
            }
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.69.153/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
```

```
                address 127.0.0.1/32;
            }
        }
    }
}
forwarding-options {
    sampling {
        traceoptions {
            file sampled world-readable;
            flag all;
        }
        input {
            rate 1;
            run-length 0;
            max-packets-per-second 20000;
        }
        family inet {
            output {
                flow-inactive-timeout 60;
                flow-active-timeout 60;
                flow-server 198.51.100.2 {
                    port 2055;
                    local-dump;
                    version 500;
                }
            }
        }
    }
}
routing-options {
[...Output Truncated...]
    autonomous-system 200;
}
protocols {
    rsvp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        label-switched-path R1toR2 {
            from 192.168.20.1;
            to 192.168.20.2;
```

```
        }
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group to_r2 {
            type internal;
            local-address 192.168.20.1;
            neighbor 192.168.20.2 {
                family inet-vpn {
                    unicast;
                }
            }
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
}
policy-options {
    policy-statement vpna-export {
        term one {
            from protocol [ bgp direct ];
            then {
                community add vpna-comm;
                accept;
            }
        }
        term two {
            then reject;
        }
    }
    policy-statement vpna-import {
        term one {
            from {
                protocol bgp;
```

```
                community vpna-comm;
            }
            then accept;
        }
        term two {
            then reject;
        }
    }
    community vpna-comm members target:200:100;
}
firewall {
    family inet {
        filter fw {
            term 1 {
                from {
                    protocol tcp;
                    port bgp;
                }
                then accept;
            }
            term 2 {
                then sample;
            }
        }
    }
}
routing-instances {
    vrf1 {
        instance-type vrf;
        interface ge-2/0/2.0;
        route-distinguisher 10.255.15.51:1;
        vrf-import vpna-import;
        vrf-export vpna-export;
        protocols {
            bgp {
                group customer {
                    type external;
                    peer-as 95000;
                    as-override;
                    neighbor 192.168.30.1;
                    neighbor 192.0.2.1;
                }
            }
        }
```

```
    }
}
```

Configuring the PE2 Router

Step-by-Step Procedure

In this step, you configure a filter with a matching term that contains the **then sample** statement and apply the filter to the ingress interface. You also configure a VRF routing instance with import and export policies. In addition, you configure interfaces, forwarding options, routing options, protocols, and policy options for the PE2 router. To configure the PE2 router:

1. Create the **fw** firewall filter that is applied to the logical interface being sampled:

   ```
   [edit firewall]
   user@router-pe2# set family inet filter fw term 1 from protocol tcp
   user@router-pe2# set family inet filter fw term 1 from port bgp
   user@router-pe2# set family inet filter fw term 1 then accept
   user@router-pe2# set family inet filter fw term 2 then sample
   user@router-pe2# set family inet filter fw term 2 then accept
   ```

2. Configure two interfaces, one interface that connects to the CE2 router (**ge-3/1/2**), and another that connects to the PE1 router (**ge-3/1/0**):

   ```
   [edit interfaces]
   user@router-pe2# set ge-3/1/0 unit 0 family inet address 192.168.20.2/24
   user@router-pe2#  set ge-3/1/0 unit 0 family mpls
   user@router-pe2#  set ge-3/1/2 unit 0 family inet address 10.10.10.2/24
   ```

3. Enable MPLS on the interface that connects to the PE1 router (**ge-3/1/0**):

   ```
   [edit interfaces]
   user@router-pe2#  set ge-3/1/0 unit 0 family mpls
   ```

4. On the interface that connects to the CE2 router, apply the **fw** filter that was configured in the firewall configuration:

   ```
   [edit interfaces]
   user@router-pe2#  set ge-3/1/2 unit 0 family inet filter input fw
   user@router-pe2#  set ge-3/1/2 unit 0 family inet filter output fw
   ```

5. Configure the **sampled** log file in the **/var/log** directory to record traffic sampling:

   ```
   [edit forwarding-options]
   ```

```
user@router-pe2# set sampling traceoptions file sampled
user@router-pe2# set sampling traceoptions file world-readable
user@router-pe1# set sampling traceoptions flag all
```

6. Specify the sampling rate and threshold value for traffic sampling:

```
[edit forwarding-options]
user@router-pe2# set sampling input rate 1
user@router-pe2# set sampling input run-length 0
user@router-pe2# set sampling input max-packets-per-second 20000
```

7. Specify active and inactive flow periods, and the router (**198.51.100.2**) that sends out the monitored information:

```
[edit forwarding-options]
user@router-pe2# set sampling family inet output flow-active-timeout 60
user@router-pe2# set sampling family inet output flow-inactive-timeout 60
user@router-pe2# set sampling family inet output flow-server 198.51.100.2 port 2055
user@router-pe2# set sampling family inet output flow-server 198.51.100.2 local-dump
user@router-pe2# set sampling family inet output flow-server 198.51.100.2 version 500
```

8. Configure the autonomous system to establish a connection between BGP peers:

```
[edit routing-options]
user@router-pe2# set autonomous-system 200
```

9. Configure RSVP to support MPLS label-switched paths (LSPs) between the PE routers:

```
[edit protocols]
user@router-pe2# set rsvp interface all
user@router-pe2# set rsvp interface fxp0.0 disable
```

10. Configure an MPLS LSP from the PE2 router to the PE1 router:

```
[edit protocols]
user@router-pe2# set mpls label-switched-path R2toR1 from 192.168.20.2
user@router-pe2# set mpls label-switched-path R2toR1 to 192.168.20.1
user@router-pe2# set mpls interface all
user@router-pe2# set mpls interface fxp0.0 disable
```

11. Configure an internal BGP group for the PE routers. Include the **family inet-vpn unicast** statement to enable BGP to carry network layer reachability information (NLRI) parameters and for BGP peers to only carry unicast routes for forwarding:

```
[edit protocols]
user@router-pe2# set bgp group to_r1 type internal
user@router-pe2# set bgp group to_r1 local-address 192.168.20.2
user@router-pe2# set bgp group to_r1 neighbor 192.168.20.1 family inet-vpn unicast
```

12. Configure OSPF as the interior gateway protocol (IGP) and to compute the MPLS LSPs:

```
[edit protocols]
user@router-pe2# set ospf traffic-engineering
user@router-pe2# set ospf area 0.0.0.0 interface all
user@router-pe2# set ospf area 0.0.0.0 interface fxp0.0 disable
```

13. Create the extended community that is applied in the policy options configuration:

```
[edit policy-options]
user@router-pe2# set community vpna-comm members target:200:100
```

14. Define the **vpna-export** routing policy that is applied in the **vrf-export** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe2# set policy-statement vpna-export term one from protocol bgp
user@router-pe2# set policy-statement vpna-export term one from protocol direct
user@router-pe2# set policy-statement vpna-export term one then community add vpna-comm
user@router-pe2# set policy-statement vpna-export term one then accept
user@router-pe2# set policy-statement vpna-export term two then reject
```

15. Define the **vpna-import** routing policy that is applied in the **vrf-import** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe2# set policy-statement vpna-import term one from protocol bgp
user@router-pe2# set policy-statement vpna-import term one from community vpna-comm
user@router-pe2# set policy-statement vpna-import term one then accept
user@router-pe2# set policy-statement vpna-import term two then reject
```

16. Configure a VRF routing instance so that routes received from the provider edge-provider edge (PE-PE) session can be imported into any of the instance's VRF secondary routing tables:

```
[edit routing-instances]
user@router-pe2# set vrf1 instance-type vrf
user@router-pe2# set vrf1 interface ge-3/1/2.0
user@router-pe2# set vrf1 route-distinguisher 10.255.19.12:1
```

```
user@router-pe2# set vrf1 vrf-import vpna-import
user@router-pe2# set vrf1 vrf-export vpna-export
user@router-pe2#set vrf1 protocols bgp group R3-R4 type external
user@router-pe2# set vrf1 protocols bgp group R3-R4 peer-as 65000
user@router-pe2# set vrf1 protocols bgp group R3-R4 as-override
user@router-pe2# set vrf1 protocols bgp group R3-R4 neighbor 10.10.10.1
```

Results

Check the results of the configuration for the PE2 router:

```
user@router-pe2> show configuration
[...Output Truncated...]
}
interfaces {
    ge-3/1/0 {
        unit 0 {
            family inet {
                address 192.168.20.2/24;
            }
            family mpls;
        }
    }
    ge-3/1/2 {
        unit 0 {
            family inet {
                filter {
                    input fw;
                    output fw;
                }
                address 10.10.10.2/24;
            }
        }
    }
}
forwarding-options {
    sampling {
        traceoptions {
            file sampled world-readable;
            flag all;
        }
        input {
            rate 1;
```

```
            run-length 0;
            max-packets-per-second 20000;
        }
        family inet {
            output {
                flow-inactive-timeout 60;
                flow-active-timeout 60;
                flow-server 198.51.100.2 {
                    port 2055;
                    local-dump;
                    version 500;
                }
            }
        }
    }
}
routing-options {
[...Output Truncated...]
    autonomous-system 200;
}
protocols {
    rsvp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        label-switched-path R2toR1 {
            from 192.168.20.2;
            to 192.168.20.1;
        }
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group to_r1 {
            type internal;
            local-address 192.168.20.2;
            neighbor 192.168.20.1 {
                family inet-vpn {
                    unicast;
```

```
                }
            }
            neighbor 192.0.2.1;
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
}
policy-options {
    policy-statement vpna-export {
        term one {
            from protocol [ bgp direct ];
            then {
                community add vpna-comm;
                accept;
            }
        }
        term two {
            then reject;
        }
    }
    policy-statement vpna-import {
        term one {
            from {
                protocol bgp;
                community vpna-comm;
            }
            then accept;
        }
        term two {
            then reject;
        }
    }
    community vpna-comm members target:200:100;
}
firewall {
    family inet {
```

```
        filter fw {
            term 1 {
                from {
                    protocol tcp;
                    port bgp;
                }
                then accept;
            }
            term 2 {
                then {
                    sample;
                    accept;
                }
            }
        }
    }
}
routing-instances {
    vrf1 {
        instance-type vrf;
        interface ge-3/1/2.0;
        route-distinguisher 10.255.19.12:1;
        vrf-import vpna-import;
        vrf-export vpna-export;
        protocols {
            bgp {
                group R3-R4 {
                    type external;
                    peer-as 65000;
                    as-override;
                    neighbor 10.10.10.1;
                }
            }
        }
    }
}
```

Configuring the CE2 Router

Step-by-Step Procedure

In this step, you configure interfaces, routing options, protocols, and policy options for the CE2 router. To configure the CE2 router:

1. Configure one interface with two IP addresses. One address is for traffic to the PE2 router and the other address is to check that traffic is flowing from the CE1 router:

```
[edit interfaces]
user@router-ce2# set ge-0/1/2 unit 0 family inet address 10.10.10.1/24
user@router-ce2#  set ge-0/1/2 unit 0 family inet address 10.4.4.4/16
```

2. Configure the autonomous system to establish a connection between BGP peers:

```
[edit routing-options]
user@router-ce1# set autonomous-system 65000
```

3. Configure BGP as the routing protocol between the CE and the PE routers:

```
[edit protocols]
user@router-ce2# set bgp group R3-R4 type external
user@router-ce2#  set bgp group R3-R4 export l3vpn-policy
user@router-ce2#  set bgp group R3-R4 peer-as 200
user@router-ce2#  set bgp group R3-R4 neighbor 10.10.10.2
```

4. Configure the policies that ensure that the CE routers exchange routing information. In this example, Router CE2 exchanges routing information with Router CE1:

```
[edit policy-options]
user@router-ce2# set policy-statement l3vpn-policy term one from protocol direct
user@router-ce2#  set policy-statement l3vpn-policy term one from route-filter 10.255.15.75/32
exact
user@router-ce2# set policy-statement l3vpn-policy term one then accept
user@router-ce2# set policy-statement l3vpn-policy term two from protocol direct
user@router-ce2# set policy-statement l3vpn-policy term two from route-filter 10.4.0.0/16 exact
user@router-ce2#  set policy-statement l3vpn-policy term two then accept
```

Results

The output below shows the configuration of the CE2 router:

```
[edit]
user@router-ce2# show
[...Output Truncated...]
```

```
interfaces {
    ge-0/1/2 {
        unit 0 {
            family inet {
                address 10.10.10.1/24;
                address 10.4.4.4/16;
            }
        }
    }
}
routing-options {
    autonomous-system 65000;
}
protocols {
    bgp {
        group R3-R4 {
            type external;
            export l3vpn-policy;
            peer-as 200;
            neighbor 10.10.10.2;
        }
    }
}
policy-options {
    policy-statement l3vpn-policy {
        term one {
            from {
                protocol direct;
                route-filter 10.255.15.75/32 exact;
            }
            then accept;
        }
        term two {
            from {
                protocol direct;
                route-filter 10.4.0.0/16 exact;
            }
            then accept;
        }
    }
}
```

Verification

IN THIS SECTION

After you have completed the configuration of the four routers, you can verify that traffic is flowing from the CE1 router to the CE2 router, and you can observe the sampled traffic from two locations. To confirm that the configuration is working properly, perform these tasks:

Verifying the Traffic Flow Between the CE Routers

Purpose

Use the **ping** command to verify traffic between the CE routers.

Action

From the CE1 router, issue the **ping** command to the CE2 router:

```
user@router-ce2> ping 10.4.4.4 source 198.51.100.2
PING 10.4.4.4 (10.4.4.4): 56 data bytes
64 bytes from 10.4.4.4: icmp_seq=0 ttl=64 time=0.861 ms
64 bytes from 10.4.4.4: icmp_seq=1 ttl=64 time=0.869 ms
64 bytes from 10.4.4.4: icmp_seq=2 ttl=64 time=0.786 ms
^C
--- 10.4.4.4 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.786/0.839/0.869/0.037 ms
```

Meaning

The output from the **ping** command shows that the **ping** command was successful. Traffic is flowing between the CE routers.

Verifying Sampled Traffic

Purpose

You can observe the sampled traffic using the **show log sampled** command from the CLI or from the router shell using the **tail –f /var/log/sampled** command. In addition, you can collect the logs in a flowcollector. The same information appears in the output of both commands and in the flow collector. For information about using a flow collector, see "Sending cflowd Records to Flow Collector Interfaces" on page 282 and "Example: Configuring a Flow Collector Interface on an M, MX or T Series Router" on page 253.

Action

From the PE1 router, use the **show log sampled** command:

```
user@router-pe1> show log sampled
[...Output Truncated...]
Nov 16 23:24:19    Src addr: 198.51.100.2
Nov 16 23:24:19    Dst addr: 10.4.4.4
Nov 16 23:24:19    Nhop addr: 192.168.20.2
Nov 16 23:24:19  Input interface: 503      # SNMP index of the incoming interface on PE1
Nov 16 23:24:19  Output interface: 505      # SNMP index of the outgoing interface on PE1
Nov 16 23:24:19    Pkts in flow: 5
Nov 16 23:24:19    Bytes in flow: 420
Nov 16 23:24:19    Start time of flow: 602411369
Nov 16 23:24:19    End time of flow: 602415369
Nov 16 23:24:19    Src port: 0
Nov 16 23:24:19    Dst port: 2048
Nov 16 23:24:19    TCP flags: 0x0
Nov 16 23:24:19    IP proto num: 1
Nov 16 23:24:19    TOS: 0x0
Nov 16 23:24:19  Src AS: 95000      # The autonomous system of CE1
Nov 16 23:24:19  Dst AS: 65000,,,,,# The autonomous system of CE2
Nov 16 23:24:19  Src netmask len: 8
Nov 16 23:24:19  Dst netmask len: 16
Nov 16 23:24:19 cflowd header:
Nov 16 23:24:19   Num-records: 1
Nov 16 23:24:19   Version: 500
Nov 16 23:24:19   Flow seq num: 13
Nov 16 23:24:19   Sys Uptime: 602450382 (msecs)
Nov 16 23:24:19   Time-since-epoch: 1258413859 (secs)
Nov 16 23:24:19   Engine id: 0
Nov 16 23:24:19   Engine type: 0
Nov 16 23:24:19   Sample interval: 1
[...Output Truncated...]
```

Meaning

The output from the **show log sampled** command shows the correct SNMP index for the incoming and outgoing interfaces on the PE1 router. Also, the source and destination addresses for the autonomous systems for the two CE routers are correct.

Cross Verifying Sampled Traffic

Purpose

You can also double check that the sampled traffic is the correct traffic by using the **show interface** ***interface-name-fpc/pic/port.unit-number*** **| match SNMP** command and the **show route** ***route-name*** **detail** command.

Action

The following output is a cross check of the output in the "Verifying Sampled Traffic" on page 107 task:

```
user@router-pe1> show interfaces ge-2/0/2.0 | match SNMP
  Logical interface ge-2/0/2.0 (Index 76) (SNMP ifIndex 503)
    Flags: SNMP-Traps 0x4000000 Encapsulation: ENET2
```

```
user@router-pe1> show route 10.4.4.4 detail

vrf1.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
10.4.0.0/16 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 10.255.19.12:1
                Next hop type: Indirect
                Next-hop reference count: 6
                Source: 192.168.20.2
                Next hop type: Router, Next hop index: 659
                Next hop: 192.168.20.2 via ge-2/0/0.0 weight 0x1, selected
                Label operation: Push 299776
                Protocol next hop: 192.168.20.2
                Push 299776
                Indirect next hop: 8e6f780 1048574
                State: <Secondary Active Int Ext>
                Local AS:   200 Peer AS:   200
                Age: 3d 19:49:32  Metric2: 65535
                Task: BGP_200.20.20.20.2+179
                Announcement bits (3): 0-RT 1-BGP RT Background 2-KRT
                AS path: 65000 I
                AS path: Recorded
                Communities: target:200:100
                Import Accepted
```

```
                VPN Label: 299776
                Localpref: 100
                Router ID: 10.10.10.2
                Primary Routing Table bgp.l3vpn.0
```

Meaning

The output of the **show interfaces ge-2/0/2.0 | match SNMP** command shows that the SNMP ifIndex field has the same value (**503**) as the output for the **show log sampled** command in the "Verifying Sampled Traffic" on page 107 task, indicating that the intended traffic is being sampled.

The output of the **show route 10.4.4.4 detail** command shows that the source address **10.4.4.4**, the source mask (**16**), and the source AS (**65000**) have the same values as the output for the **show log sampled** command in the "Verifying Sampled Traffic" on page 107 task, indicating that the intended traffic is being sampled.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

Configuring an Aggregate Export Timer on M, MX and T Series Routers for Version 8 Records

When you use flow monitoring version 8 records for active flow monitoring, you can configure an aggregate export timer. To configure this timer, include the **aggregate-export-interval** statement at the **[edit forwarding-options sampling output]** hierarchy level. The timer value has a default minimum setting of 90 seconds and a maximum value of 1800 seconds.

```
[edit]
forwarding-options {
    sampling {
        output {
            aggregate-export-interval duration;
        }
    }
}
```

Option: Configuring Port Mirroring with Filter-Based Forwarding and a Monitoring Group

For active flow monitoring, you can load-balance traffic across multiple Monitoring Services PICs using the same method as passive flow monitoring. The only difference is that you do not configure the input interface with the **passive-monitor-mode** statement at the **[edit interfaces *interface-name*]** hierarchy level.

To load-balance traffic for active flow monitoring, port-mirror the incoming packets to a tunnel services interface. Redirect this copy of the traffic to a filter-based forwarding instance by applying a firewall filter to the tunnel services interface. Configure the instance to send the traffic to a group of monitoring services interfaces. Finally, use a monitoring group to send flow records from the monitoring services interfaces to a flow server.

> **NOTE:** When you load-balance port-mirrored traffic across several Monitoring Services interfaces, there are some limitations:
>
> - The original Monitoring Services PIC supports this method. You cannot use a Monitoring Services II PIC.
> - You must use the suite of **show passive-monitoring** commands to monitor traffic. The **show services accounting** commands are not supported.
> - Because load-balanced traffic is routed through the Tunnel Services PIC, the total throughput of the load-balanced traffic coming from the Monitoring Services PICs cannot exceed the bandwidth of the tunnel interface.

For detailed information on this method, see "Copying and Redirecting Traffic with Port Mirroring and Filter-Based Forwarding" on page 229.

Sending Port-Mirrored Traffic from an M, MX or T Series Router to Multiple Export Interfaces by Using Next-Hop Groups

To send port-mirrored traffic to multiple flow servers or packet analyzers, you can use the **next-hop-group** statement. The router can make up to 16 copies of traffic per group and send the traffic to the next-hop group members you configure. 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 configure multiple port mirroring with next-hop groups, include the **next-hop-group** statement at the **[edit forwarding-options]** hierarchy level.

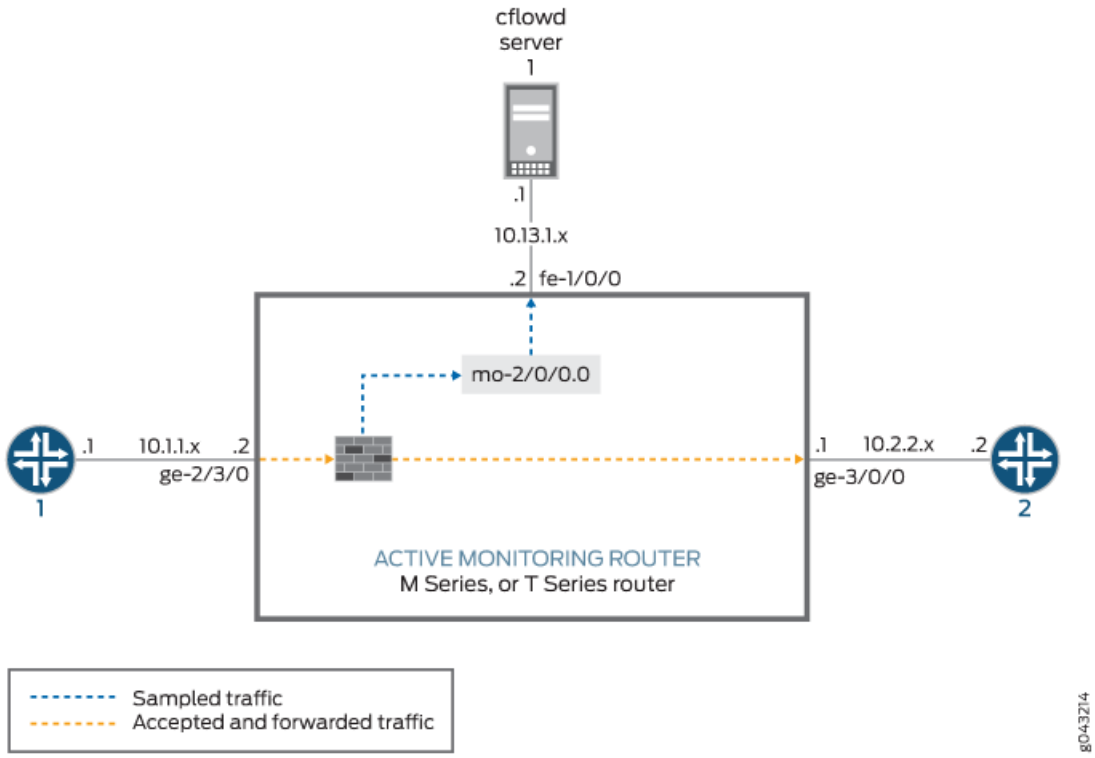
You must port-mirror the initial traffic to a tunnel interface so that it can be filtered and duplicated. Also, you need configure only the interface names for point-to-point interfaces, but you must configure the interface names and a next hop for multipoint interfaces (such as Ethernet).

```
[edit]
forwarding-options {
    port-mirroring {
        family inet {
            input {
                rate 1;
            }
            output {
                interface vt-3/3/0.1;
                no-filter-check;
            }
        }
    }
    next-hop-group ftp-traffic {
        interface so-4/3/0.0;
        interface so-0/3/0.0;
    }
    next-hop-group http-traffic {
        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;
    }
}
```

NOTE: **Next-hop groups** are supported on M Series routers only, except the M120 router and the M320 router.

Example: Sampling Configuration for M, MX and T Series Routers

Figure 17: Active Flow Monitoring—Sampling Configuration Topology Diagram



In Figure 17 on page 113, traffic from Router 1 arrives on the monitoring router's Gigabit Ethernet **ge-2/3/0** interface. The exit interface on the monitoring router that leads to destination Router 2 is **ge-3/0/0**. In active flow monitoring, both the input interface and exit interface can be any interface type (such as SONET/SDH, Gigabit Ethernet, and so on). The export interface leading to the flow server is **fe-1/0/0**.

Configure a firewall filter to sample, count, and accept all traffic. Apply the filter to the input interface, and configure the exit interface (for traffic forwarding), the adaptive services interface (for flow processing), and the export interface (for exporting flow records).

Configure sampling at the **[edit forwarding-options]** hierarchy level. Include the IP address and port of the flow server with the **flow-server** statement and specify the adaptive services interface to be used for flow record processing with the **interface** statement at the **[edit forwarding-options sampling]** hierarchy level.

Router 1

```
[edit]
interfaces {
    sp-2/0/0 { # This adaptive services interface creates the flow records.
        unit 0 {
            family inet {
                address 10.5.5.1/32 {
                    destination 10.5.5.2;
                }
            }
        }
    }
    fe-1/0/0 { # This is the interface where records are sent to the flow server.
        unit 0 {
            family inet {
                address 10.60.2.2/30;
            }
        }
    }
    ge-2/3/0 { # This is the input interface where all traffic enters the router.
        unit 0 {
            family inet {
                filter {
                    input catch_all; # This is where the firewall filter is applied.
                }
                address 10.1.1.1/20;
            }
        }
    }
    ge-3/0/0 { # This is the interface where the original traffic is forwarded.
        unit 0 {
            family inet {
                address 10.2.2.1/24;
            }
        }
    }
}
forwarding-options {
    sampling { # Traffic is sampled and sent to a flow server.
        input {
            rate 1; # Samples 1 out of  x  packets (here, a rate of 1 sample per packet).
            }
        }
```

```
        family inet {
            output {
                flow-server 10.60.2.1 { # The IP address and port of the flow server.
                    port 2055;
                    version 5; # Records are sent to the flow server using version 5 format.
                }
                flow-inactive-timeout 15;
                flow-active-timeout 60;
                interface sp-2/0/0 { # Adding an interface here enables PIC-based sampling.
                    engine-id 5; # Engine statements are dynamic, but can be configured.
                    engine-type 55;
                    source-address 10.60.2.2; # You must configure this statement.
                }
            }
        }
    }
    firewall {
        family inet {
            filter catch_all { # Apply this filter on the input interface.
                term default {
                    then {
                        sample;
                        count counter1;
                        accept;
                    }
                }
            }
        }
    }
```

Verifying Your Work

To verify that your configuration is correct, use the following commands on the monitoring station that is configured for active flow monitoring:

- **show services accounting errors**
- **show services accounting (flow | flow-detail)**
- **show services accounting memory**
- **show services accounting packet-size-distribution**
- **show services accounting status**

- **show services accounting usage**
- **show services accounting aggregation template template-name *name* (detail | extensive | terse)** (version 9 only)

Most active flow monitoring operational mode commands contain equivalent output information to the following passive flow monitoring commands:

- **show services accounting errors = show passive-monitoring error**
- **show services accounting flow = show passive-monitoring flow**
- **show services accounting memory = show passive-monitoring memory**
- **show services accounting status = show passive-monitoring status**
- **show services accounting usage = show passive-monitoring usage**

The active flow monitoring commands can be used with most active flow monitoring applications, including sampling, discard accounting, port mirroring, and multiple port mirroring. However, you can use the passive flow monitoring commands only with configurations that contain a monitoring group at the **[edit forwarding-options monitoring]** hierarchy level.

The following shows the output of the **show** commands used with the configuration example:

```
user@router1> show services accounting errors
```

```
Service Accounting interface: sp-2/0/0, Local interface index: 542
Service name: (default sampling)
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory overload: No, PPS overload: No, BPS overload: Yes

user@router1> show services accounting flow-detail limit 10
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
Protocol   Source         Source  Destination Destination    Packet        Byte
           Address          Port  Address            Port     count       count
udp(17)    10.1.1.2           53  10.0.0.1             53      4329     3386035
ip(0)      10.1.1.2            0  10.0.0.2              0      4785     3719654
ip(0)      10.1.1.2            0  10.0.1.2              0      4530     3518769
udp(17)    10.1.1.2            0  10.0.7.1              0      5011     3916767
tcp(6)     10.1.1.2           20  10.3.0.1             20         1        1494
tcp(6)     10.1.1.2           20  10.168.80.1          20         1         677
tcp(6)     10.1.1.2           20  10.69.192.1          20         1         446
```

```
tcp(6)     10.1.1.2           20  10.239.240.1         20         1        1426
tcp(6)     10.1.1.2           20  10.126.160.1         20         1         889
tcp(6)     10.1.1.2           20  10.71.224.1          20         1        1046

user@router1>  show services accounting memory
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
  Memory utilization
    Allocation count: 437340, Free count: 430681, Maximum allocated: 6782
    Allocations per second: 3366, Frees per second: 6412
   Total memory used (in bytes): 133416928, Total memory free (in bytes): 133961744

user@router1>  show services accounting packet-size-distribution
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
Range start     Range end     Number of packets     Percentage packets
         64            96               1705156                    100

user@router1>  show services accounting status
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
  Interface state: Monitoring
  Group index: 0
  Export interval: 60 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 55, Engine ID: 5
  Route record count: 13, IFL to SNMP index count: 30, AS count: 1
  Time set: Yes, Configuration set: Yes
  Route record set: Yes, IFL SNMP map set: Yes

user@router1>  show services accounting usage
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
  CPU utilization
    Uptime: 4790345 milliseconds, Interrupt time: 1668537848 microseconds
    Load (5 second): 71%, Load (1 minute): 63%
```

Associating Sampling Instances for Active Flow Monitoring with a Specific FPC, MPC, or DPC

The Junos OS enables you to configure sampling instances for active flow monitoring, by specifying a name for the sampling parameters and associating the instance name with a specific FPC, MPC, or DPC.

To configure active sampling instances, include the **instance** statement at the **[edit forwarding-options sampling]** hierarchy level. For more information about configuring sampling instances, see the *Junos OS Services Interfaces Library for Routing Devices*.

To associate a configured active sampling instance with a specific FPC, MPC, or DPC, include the sampling instance name at the **[edit chassis fpc *slot-number*]** hierarchy level:

```
[edit chassis fpc slot-number]
sampling-instance instance-name;
```

On a TX Matrix, TX Matrix Plus router, include the **sampling-instance** statement at the **[edit chassis lcc *number* fpc *slot-number*]** hierarchy level:

```
[edit chassis lcc number fpc slot-number]
sampling-instance instance-name;
```

RELATED DOCUMENTATION

Example: Sampling Instance Configuration | 118

sampling-instance | 1130

Example: Sampling Instance Configuration

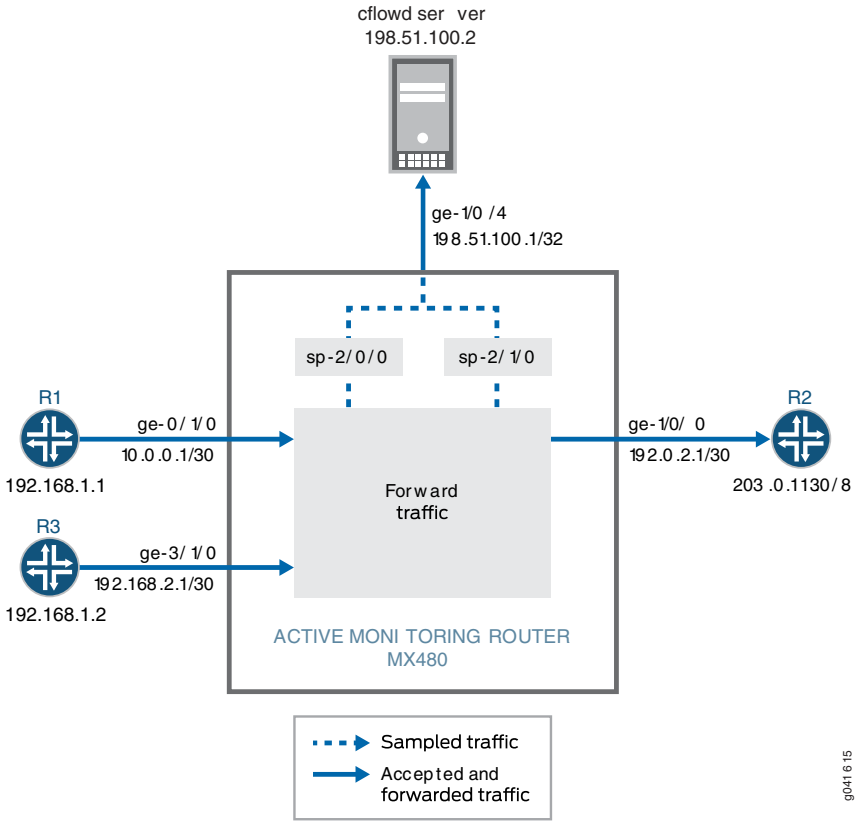
IN THIS SECTION

You can configure active sampling using a sampling instance and associate that sampling instance to a particular FPC, MPC, or DPC. In addition, you can define multiple sampling instances associated with multiple destinations and protocol families per sampling instance destination.

Example Network Details

The following example shows the configuration of two sampling instances on an MX480 router running Junos OS Release 9.6.

Figure 18: Active Flow Monitoring—Sampling Instance Configuration Topology Diagram



In Figure 18 on page 119, packets from Router 1 arrive on the monitoring router's Gigabit Ethernet **ge-0/1/0** interface, the packets are sampled by the services interface **sp-2/0/0** and sent to the cflowd server by the export interface **ge-1/0/4**. Packets from Router 3 arrive on the monitoring router's Gigabit Ethernet **ge-3/1/0** interface, the packets are sampled by the services interface **sp-2/1/0** and sent to the cflowd server by the export interface **ge-1/0/4**. Normal traffic flow from **ge-0/1/0** and **ge-3/1/0** to **ge-1/0/0** and on to Router 2 continues undisturbed during the sampling process. In active flow monitoring, both the input interface and exit interface can be any interface type (such as SONET/SDH, Gigabit Ethernet, and so on).

Only one sampling instance can be attached to an FPC, MPC, or DPC. Multiple families can be configured under a sampling instance. Each family can have its own collector address. You can define sampling instances and attach each instance to different FPCs, or a single sampling instance can be attached to all FPCs.

The sampling configuration for this example includes the following:

- Two sampling instances, **s0** and **s1**, configured to collect sampling data at the **[edit forwarding-options]** hierarchy level. The **flow-server** statement includes the IP address, port, and template of the flow server. The **interface** statement includes the services interface, **sp-2/0/0** or **sp-2/1/0**, for flow record processing, and the source address of the incoming router on the sampled interface.
- The binding of the two sampling instances to FPCs 0 and 3. These are configured with the **sampling-instance** statement at the **[edit chassis fpc *slot*]** hierarchy level.
- Sampling activated on the input interfaces **ge-0/1/0** and **ge-3/1/0** using the **sampling** statement at the **[edit interfaces *interface-name* unit *unit-number* family *family*]** hierarchy level.

In this example, the **ping** command is issued on Router 1 to Router 2 via the MX480 router to generate traffic. After the packets are generated, **show** commands are issued to verify that the sampling configuration is working as expected.

Example Router Configuration

The following output shows the configuration of an MX480 router with two sampling instances.

```
user@MX480–router> show configuration
[...Output Truncated...]
}
chassis {
    fpc 0 { # The fpc number is associated with the interface on which sampling
is enabled, ge-0/1/0 in this statement.
        sampling-instance s0;
    }
    fpc 3 { # The fpc number is associated with the interface on which sampling
is enabled, ge-3/1/0 in this statement.
        sampling-instance s1;
    }
}
interfaces {
    ge-0/1/0 { # This interface has sampling activated.
        unit 0 {
            family inet {
                sampling { # Here sampling is activated.
                    input;
                }
                address 10.0.0.1/30;
```

```
            }
        }
    }
    ge-1/0/0 { # The interface on which packets are exiting the router.
        unit 0 {
            family inet {
                address 192.0.2.1/30;
            }
        }
    }
    ge-1/0/4 { # The interface connected to the cflowd server.
        unit 0 {
            family inet {
                address 198.51.100.1/32;
            }
        }
    }
    sp-2/0/0 { # The service interface that samples the packets from Router 1.
        unit 0 {
            family inet;
        }
    }
    sp-2/1/0 { # The service interface that samples the packets from Router 3.
        unit 0 {
            family inet;
        }
    }
    ge-3/1/0 { # This interface has sampling activated.
        unit 0 {
            family inet {
                sampling { # Here sampling is activated.
                    input;
                }
                address 192.168.2.1/30;
            }
        }
    }
}
forwarding-options {
    sampling {
        instance {
            s0 {
                input {
                    rate 1;
```

```
                    run-length 0;
                }
                family inet {
                    output {
                        flow-server 198.51.100.2 { # The address of the external
server.
                            port 2055;
                            version9 {
                                template {
                                    v4
                                }
                            }
                        }
                        interface sp-2/0/0 {
                            source-address 192.168.1.1; # Source address of the
sampled packets
                        }
                    }
                }
            }
            s1 {
                input {
                    rate 1;
                    run-length 0;
                }
                family inet {
                    output {
                        flow-server 198.51.100.2 { # The address of the external
server.
                            port 2055;
                            version9 {
                                template {
                                    v4
                                }
                            }
                        }
                        interface sp-2/1/0 {
                            source-address 192.168.1.2; # Source address of the
sampled packets
                        }
                    }
                }
            }
        }
```

```
    }
}

routing-options {
    static {
        route 203.0.113.0/8 next-hop 192.0.2.2;
    }
}
services {
    flow-monitoring {
        version9 {
            template v4 {
                flow-active-timeout 30;
                flow-inactive-timeout 30;
                ipv4-template;
            }
        }
    }
}
```

Configuration Commands Used for the Configuration Example

The following **set** commands are used for the configuration of the sampling instance in this example. Replace the values in these commands with values relevant to your own network.

- **set chassis fpc 0 sampling-instance s0**
- **set chassis fpc 3 sampling-instance s1**
- **set interfaces ge-0/1/0 unit 0 family inet sampling input**
- **set interfaces ge-0/1/0 unit 0 family inet address**
- **set interfaces ge-1/0/0 unit 0 family inet address**
- **set interfaces sp-2/0/0 unit 0 family inet**
- **set interfaces sp-2/1/0 unit 0 family inet**
- **set interfaces ge-3/1/0 unit 0 family inet sampling input**
- **set interfaces ge-3/1/0 unit 0 family inet address**
- **set forwarding-options sampling instance s0 input rate 1**
- **set forwarding-options sampling instance s0 input run-length 0**
- **set forwarding-options sampling instance s0 family inet output flow-server 198.51.100.2 port 2055**

- **set forwarding-options sampling instance s0 family inet output flow-server 198.51.100.2 version9 template v4;**
- **set forwarding-options sampling instance s0 family inet output interface sp-2/0/0 source-address 192.168.1.1**
- **set forwarding-options sampling instance s1 input rate 1**
- **set forwarding-options sampling instance s1 input run-length 0**
- **set forwarding-options sampling instance s1 family inet output flow-server 198.51.100.2 port 2055**
- **set forwarding-options sampling instance s1 family inet output flow-server 198.51.100.2 version9 template v4;**
- **set forwarding-options sampling instance s1 family inet output interface sp-2/1/0 source-address 192.168.1.2**
- **set routing-options static route 203.0.113.0/8 next-hop 192.0.2.2**
- **set services flow-monitoring version9 template v4 flow-active-timeout 30**
- **set services flow-monitoring version9 template v4 flow-inactive-timeout 30**
- **set services flow-monitoring version9 template v4 ipv4-template**

Verifying Your Work

To verify that your configuration is working as expected, use the following commands on the router that is configured with the sampling instance:

- **show services accounting aggregation template template-name *template-name***
- **show services accounting flow**

The following shows the output of the **show** commands issued on the MX480 router used in this configuration example:

```
user@MX480-router> show services accounting aggregation template template-name v4
                                 Src   Dst
                                Port/ Port/
Source          Destination      ICMP  ICMP                          Packet
Address         Address          Type  Code  Proto TOS                Count
10.0.0.6        203.0.113.3       100  1000     17   8                   14
10.0.0.5        203.0.113.2       100  1000     17   8                   15
10.0.0.3        203.0.113.3       100  1000     17   8                   15
10.0.0.2        203.0.113.3       100  1000     17   8                   15
10.0.0.4        203.0.113.2       100  1000     17   8                   15
10.0.0.6        203.0.113.2       100  1000     17   8                   15
```

```
10.0.0.4        203.0.113.3       100  1000     17   8                  15
10.0.0.2        203.0.113.2       100  1000     17   8                  16
10.0.0.3        203.0.113.2       100  1000     17   8                  15
10.0.0.5        203.0.113.3       100  1000     17   8                  15

user@MX480-router> show services accounting aggregation template template-name v4
                                Src   Dst
                               Port/ Port/
Source          Destination     ICMP  ICMP                          Packet
Address         Address         Type  Code  Proto TOS                Count
10.0.0.6        203.0.113.3     100  1000     17   8                  16
10.0.0.5        203.0.113.2     100  1000     17   8                  17
10.0.0.3        203.0.113.3     100  1000     17   8                  16
10.0.0.2        203.0.113.3     100  1000     17   8                  16
10.0.0.4        203.0.113.2     100  1000     17   8                  17
10.0.0.6        203.0.113.2     100  1000     17   8                  17
10.0.0.4        203.0.113.3     100  1000     17   8                  16
10.0.0.2        203.0.113.2     100  1000     17   8                  17
10.0.0.3        203.0.113.2     100  1000     17   8                  17
10.0.0.5        203.0.113.3     100  1000     17   8                  16

user@MX480-router> show services accounting flow
  Flow information
    Interface name: sp-2/0/0, Local interface index: 152
    Flow packets: 884, Flow bytes: 56576
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 628
    Active flows: 10, Total flows: 35
    Flows exported: 75, Flows packets exported: 14
    Flows inactive timed out: 25, Flows active timed out: 75

user@MX480-router> show services accounting flow
  Flow information
    Interface name: sp-2/0/0, Local interface index: 152
    Flow packets: 898, Flow bytes: 57472
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 628
    Active flows: 10, Total flows: 35
    Flows exported: 75, Flows packets exported: 14
    Flows inactive timed out: 25, Flows active timed out: 75
```

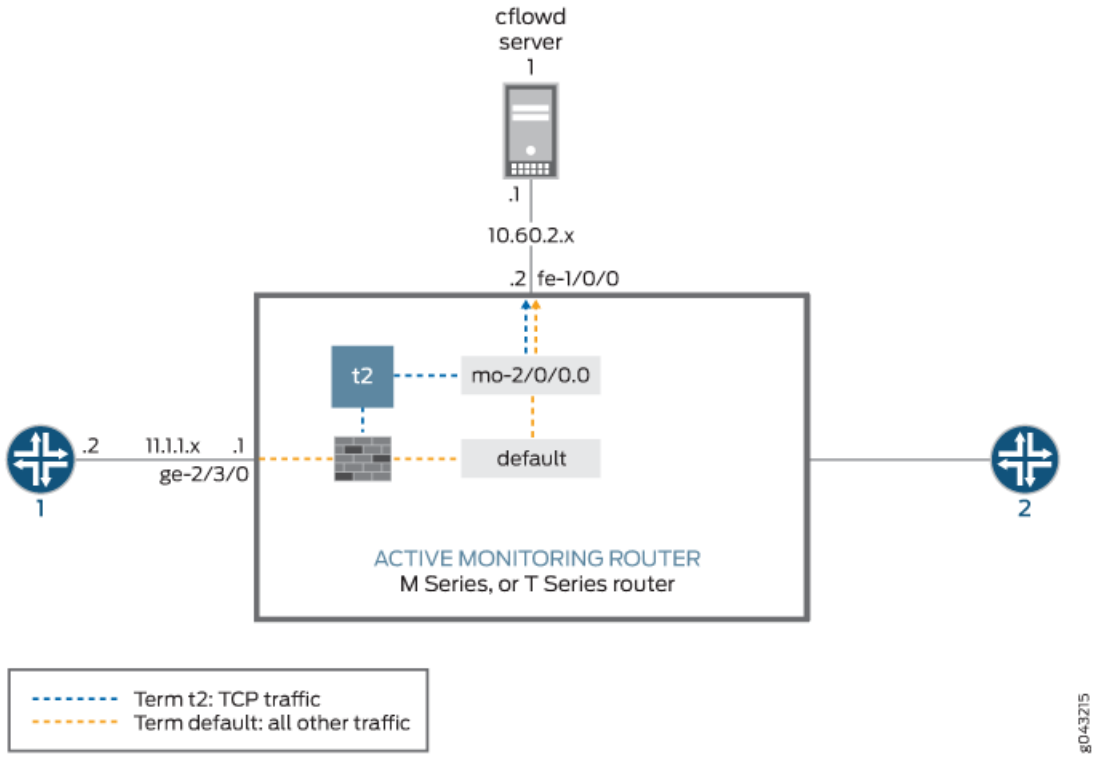
RELATED DOCUMENTATION

Configuring Sampling Instance on MX, M and T Series Routers or QFX Series Switches | 388
Configuring Active Flow Monitoring | 34
sampling-instance | 1130

Example: Sampling and Discard Accounting Configuration on M, MX and T Series Routers

Discard accounting allows you to sample traffic, send it to a flow server for analysis, and discard all packets without forwarding them to their intended destination. Discard accounting is enabled with the **discard accounting *group-name*** statement in a firewall filter at the **[edit firewall family inet filter *filter-name* term *term-name* then]** hierarchy level. Then, the filter is applied to an interface with the **filter** statement at the **[edit interfaces *interface-name* unit *unit-number* family inet]** hierarchy level and processed with the **output** statement at the **[edit forwarding-options accounting *group-name*]** hierarchy level.

Figure 19: Active Flow Monitoring—Sampling and Discard Accounting Topology Diagram



In Figure 19 on page 126, traffic from Router 1 arrives on the monitoring router's Gigabit Ethernet **ge-2/3/0** interface. The export interface leading to the flow server is **fe-1/0/0** and there is no exit interface.

In this example, TCP traffic is sent to one accounting group and all other traffic is diverted to a second group. After being sampled and counted, the two types of traffic are acted upon by the sampling and accounting processes. These processes create flow records and send the records to the version 8 flow server for analysis. Because multiple types of traffic are sent to the same server, we recommend that you configure the **engine-id**, **engine-type**, and **source-address** statements manually in your accounting and sampling hierarchies. This way, you can differentiate between traffic types when they arrive at the flow server.

```
[edit]
interfaces {
    sp-2/0/0 { # This adaptive services interface creates the flow records.
        unit 0 {
            family inet {
                address 10.5.5.1/32 {
                    destination 10.5.5.2;
                }
            }
        }
    }
    fe-1/0/0 { # This is the interface where records are sent to the flow server.
        unit 0 {
            family inet {
                address 10.60.2.2/30;
            }
        }
    }
    ge-2/3/0 { # This is the input interface where traffic enters the router.
        unit 0 {
            family inet {
                filter {
                    input catch_all;
                }
                address 10.1.1.1/20;
            }
        }
    }
}
forwarding-options {
    sampling { # The router samples the traffic.
        input {
            rate 100; # One out of every 100 packets is sampled.
            }
        }
            family inet {
```

```
        output { # The sampling process creates and exports flow records.
        flow-server 10.60.2.1 { # You can configure a variety of settings.
          port 2055;
          version 8;
          aggregation { # Aggregation is unique to flow version 8.
            protocol-port;
            source-destination-prefix;
          }
        }
        aggregate-export-interval 90;
        flow-inactive-timeout 60;
        flow-active-timeout 60;
        interface sp-2/0/0 { # This statement enables PIC-based sampling.
          engine-id 5; # Engine statements are dynamic, but can be configured.
          engine-type 55;
          source-address 10.60.2.2; # You must configure this statement.
        }
      }
}
accounting counter1 { # This discard accounting process handles default traffic.
  output { # This process creates and exports flow records.
    flow-inactive-timeout 65;
    flow-active-timeout 65;
    flow-server 10.60.2.1 { # You can configure a variety of settings.
      port 2055;
      version 8;
      aggregation { # Aggregation is unique to version 8.
        protocol-port;
        source-destination-prefix;
      }
    }
    interface sp-2/0/0 {  # This statement enables PIC-based discard accounting.
      engine-id 1; # Engine statements are dynamic, but can be configured.
      engine-type 11;
      source-address 10.60.2.3; # You must configure this statement.
    }
  }
}
accounting t2 { # The second discard accounting process handles the TCP traffic.
  output { # This process creates and exports flow records.
    aggregate-export-interval 90;
    flow-inactive-timeout 65;
    flow-active-timeout 65;
    flow-server 10.60.2.1 { # You can configure a variety of settings for the server.
```

```
            port 2055;
            version 8;
            aggregation { # Aggregation is unique to version 8.
              protocol-port;
              source-destination-prefix;
            }
          }
          interface sp-2/0/0 { # This statement enables PIC-based discard accounting.
            engine-id 2; # Engine statements are dynamic, but can be configured.
            engine-type 22;
            source-address 10.60.2.4;# You must configure this statement.
          }
        }
      }
    }
    firewall {
      family inet {
        filter catch_all { # Apply the firewall filter on the input interface.
          term t2 { # This places TCP traffic into one group for sampling and
            from { # discard accounting.
              protocol tcp;
            }
            then {
              count c2;# The count action counts traffic as it enters the router.
              sample; # The sample action sends the traffic to the sampling process.
              discard accounting t2; # The discard accounting discards traffic.
            }
          }
          term default { # Performs sampling and discard accounting on all other traffic.
            then {
              count counter; # The count action counts traffic as it enters the router.
              sample# The sample action sends the traffic to the sampling process.
              discard accounting counter1; # This activates discard accounting.
            }
          }
        }
      }
    }
```

Verifying Your Work

To verify that your configuration is correct, use the following commands on the monitoring station that is configured for active flow monitoring:

- **show services accounting aggregation** (for version 8 flows only)
- **show services accounting errors**
- **show services accounting (flow | flow-detail)**
- **show services accounting memory**
- **show services accounting packet-size-distribution**
- **show services accounting status**
- **show services accounting usage**

The following shows the output of the **show** commands used with the configuration example:

```
user@host> show services accounting flow name t2
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: t2
  Flow information
    Flow packets: 56130820, Flow bytes: 3592372480
    Flow packets 10-second rate: 13024, Flow bytes 10-second rate: 833573
    Active flows: 600, Total flows: 600
    Flows exported: 28848, Flows packets exported: 960
    Flows inactive timed out: 0, Flows active timed out: 35400

user@host> show services accounting
Service Name:
    (default sampling)
    counter1
    t2

user@host> show services accounting aggregation protocol-port detail name t2
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: t2

  Protocol: 6, Source port: 20, Destination port: 20
  Start time: 442794, End time: 6436260
  Flow count: 1, Packet count: 4294693925, Byte count: 4277471552

user@host> show services accounting aggregation source-destination-prefix name
t2 limit 10 order packets
```

```
Service Accounting interface: sp-2/0/0, Local interface index: 542
Service name: t2
     Source     Destination  Input SNMP  Output SNMP   Flow  Packet    Byte
     Prefix          Prefix       Index        Index  count   count   count
10.1.1.2/20    10.225.0.1/0          24           26      0      13    9650
10.1.1.2/20   10.143.80.1/0          24           26      0      13   10061
10.1.1.2/20   10.59.176.1/0          24           26      0      13   10426
10.1.1.2/20     10.5.32.1/0          24           26      0      13   12225
10.1.1.2/20    10.36.16.1/0          24           26      0      13    9116
10.1.1.2/20     10.1.96.1/0          24           26      0      12   11050
10.1.1.2/20    10.14.48.1/0          24           26      0      13   10812
10.1.1.2/20   10.31.192.1/0          24           26      0      13   11473
10.1.1.2/20  10.129.144.1/0          24           26      0      13    7647
10.1.1.2/20  10.188.160.1/0          24           26      0      13   10056
```

user@host> **show services accounting aggregation source-destination-prefix name**

```
t2 extensive limit 3
Service Accounting interface: sp-2/0/0, Local interface index: 542
Service name: t2

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 10.200.176.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 5340

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 10.243.160.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 5490

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 10.162.160.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 4079
```

Configuring Active Flow Monitoring on PTX Series Packet Transport Routers

You can use flow monitoring to help with network administration. Active flow monitoring on PTX Series routers allows you to collect sampled packets, then the router does GRE encapsulation of the packets and sends them to a remote server for flow processing. The GRE encapsulation includes an interface index and GRE key field. The GRE encapsulation removes MPLS tags. You configure one or more port-mirroring instances to define which traffic to sample and configure a server to receive the GRE encapsulated packets. You configure a firewall filter on interfaces where you want to capture flows. You can configure as many as 48 port-mirroring instances.

To configure the router to do GRE encapsulation of sampled packets and send them to a remote server for flow processing:

1. Configure one or more server profiles that specify a host where GRE encapsulated sampled packets are sent, and optionally, a source address to include in the header of each sampled packet.

   a. Specify a name for each server profile and an IP address of the host where sampled packets are sent:

   ```
   [edit services hosted-services]
   user@host# set server-profile server-profile-name server-address ipv4-address
   ```

   b. (Optional) For each server profile, specify a source address to include in the header of each sampled packet:

   ```
   [edit services hosted-services server-profile server-profile-name]
   user@host# set client-address ipv4-address
   ```

   NOTE: The default client address is 0.0.0.0. You must specify an IPv4 address as the client address. You can also specify the loopback address or management interface address as the client address.

2. Configure one or more port-mirroring instances.

   a. Specify a name for each port-mirroring instance:

   ```
   [edit forwarding-options port-mirroring]
   user@host# set instance instance-name
   ```

NOTE: You can configure a maximum of 48 port-mirroring instances.

b. Specify a protocol family for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name]
user@host# set family (inet | inet6 )
```

3. To set the ratio of the number of packets to sample, specify a value from 1 through 65,535 for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name input]
user@host# set rate number
```

NOTE: You must specify a value for the **rate** statement. The default value is zero, which effectively disables sampling. If, for example, you specify a rate value of 4, every fourth packet (1 packet out of 4) is sampled.

4. (Optional) Specify the number of samples to collect after the initial trigger event for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name input]
user@host# set run-length number
```

NOTE: The default value is zero. You can specify a number up to 20.

5. To designate a host where sampled traffic is sent, specify the name of server profile configured at the **[edit services hosted-services]** hierarchy level for each port-mirroring instance:

```
[edit forwarding-options port-mirroring instance instance-name family ( inet | inet6) output]
user@host# set server-profile server-profile-name
```

6. Configure one or more firewall filters.

   a. For each firewall filter, specify a protocol family, filter name, and match conditions:

      ```
      [edit firewall]
      user@host# set filter family (inet | inet6) filter filter-name term term-name from match-condtions
      ```

   b. For each firewall filter you configure, specify the name of a port-mirroring instance you configured at the **[edit forwarding-options]** hierarchy level as a nonterminating action so that the traffic that matches that instance is sampled:

      ```
      [edit firewall family (inet | inet6) filter filter-name term term-name]
      user@host# set then port-mirroring instance instance-name
      ```

7. Apply each firewall filter to an interface to evaluate incoming traffic:

   ```
   [edit interfaces interface-name unit logical-unit-number]
   user@host# set family (inet | inet6) filter input firewall-filter-name
   ```

   NOTE: Active flow monitoring is supported only on incoming traffic. You cannot apply firewall filters to evaluate outgoing traffic.

8. Configure the remote server, where GRE encapsulated packets are sent, to perform flow processing.

RELATED DOCUMENTATION

- *Configuring Port Mirroring*
- *hosted-services*
- *port-mirroring*
- *server-profile (Active Flow Monitoring)*
- *Firewall Filter Nonterminating Actions*

Configuring Actively Monitored Interfaces on M, MX and T Series Routers

Configure the input interfaces and apply the firewall filter that you defined earlier. Unlike passive flow monitoring, the input interfaces for active flow monitoring are not restricted, so you can select most standard network interfaces (such as ATM1 or Ethernet-based interfaces) as the input.

If you configure active flow monitoring with sampling, you can configure an interface filter in place of a firewall filter with the **sampling** statement at the **[edit interfaces *interface-name-fpc/pic/port* unit *unit-number* family inet]** hierarchy level.

```
[edit]
interfaces {
   so-2/2/0 {
      unit 0 {
         family inet {
            filter {
               input active_filter;
            }
            address 10.36.11.2/32 {
               destination 10.36.11.1;
            }
            sampling {
               (input | output | [input output]);
            }
         }
      }
   }
}
```

Collecting Flow Records

Traffic flows can be exported in flow monitoring version 5, 8, and 9 formats for active flow monitoring. The default export format for flow monitoring records is version 5. To change the export format to flow monitoring version 8, include the **version 8** statement at either the [**edit forwarding-options accounting *name* output flow-server *flow-server-address***] or the [**edit forwarding-options sampling output flow-server *flow-server-address***] hierarchy level. To change the export format to flow monitoring version 9, include the **version9 template *template-name*** statement at the [**edit forwarding-options sampling output flow-server *flow-server-address***] hierarchy level. For more information on flow record formats, see “Flow Monitoring Output Formats” on page 9.

To capture flow data generated by the Monitoring Services PIC, Adaptive Services PIC, or MultiServices PIC and export it to a flow server, you can use one of the following active flow monitoring methods:

- Configuring M, MX and T Series Routers for Discard Accounting with a Sampling Group on page 78
- Configuring M, MX and T Series Routers for Discard Accounting with an Accounting Group on page 77
- Configuring M, MX and T Series Routers for Discard Accounting with a Template on page 79
- Replicating M, MX and T Series Routing Engine-Based Sampling to Multiple Flow Servers on page 83
- Replicating Version 9 Flow Aggregation From M, MX and T Series Routers to Multiple Flow Servers on page 84
- Configuring Routing Engine-Based Sampling on M, MX and T Series Routers for Export to Multiple Flow Servers on page 85
- Configuring an Aggregate Export Timer on M, MX and T Series Routers for Version 8 Records on page 110

Configuring M, MX and T Series Routers for Discard Accounting with an Accounting Group

To perform discard accounting on specified traffic, you can collect flow records with the **accounting** statement at the **[edit forwarding-options]** hierarchy level. Like sampling, your topology must be simple (for example, one input interface and one export interface).

Again, you can collect flow records by specifying input and output interfaces. You can configure the input interface to perform discard accounting by applying a firewall filter that contains the **then discard accounting** statement. This match condition directs the filtered traffic to be converted into flow records and exported for analysis by the monitoring services or adaptive services interface. The original packets are then sent to the discard process. For the output, remember to specify the IP address and port of your flow server and the services interface you plan to use for processing flow records.

You must configure a source address, but the **engine-id** and **engine-type** output interface statements are added automatically. You can override these values manually to track different flows with a single flow collector. SNMP input and output interface index information is captured in flow records by default when you configure discard accounting.

```
[edit]
forwarding-options {
    accounting counter1 {
        output {
            flow-inactive-timeout 65;
            flow-active-timeout 65;
```

```
        flow-server 10.60.2.1 {
            port 2055;
            version 8;
            aggregation {
                protocol-port;
                source-destination-prefix;
            }
        }
        interface sp-2/0/0 {
            engine-id 1;
            engine-type 11;
            source-address 10.60.2.2;
        }
    }
  }
}
```

Configuring M, MX and T Series Routers for Discard Accounting with a Sampling Group

If your needs for active flow monitoring are simple, you can collect flow records with a sampling group. Sampling does not require you to configure a monitoring group (as required in passive flow monitoring) because you can configure flow server information in the **sampling** hierarchy. When you wish to sample traffic, include the **sampling** statement at the **[edit forwarding-options]** hierarchy level.

The typical sampling configuration has one input interface and one export interface. The input interface is activated by the **then sample** statement in a firewall filter term. This match condition directs traffic to the sampling process. Alternatively, you can use an interface-based filter in place of a firewall filter if you include the **sampling** statement at the **[edit interfaces *interface-name-fpc/pic/port* unit *unit-number* family inet]** hierarchy level.

There are two types of sampling available: PIC-based sampling and Routing Engine-based sampling. PIC-based sampling occurs when a monitoring services or adaptive services interface is the target for the output of the sampling process. To enable PIC-based sampling, include the **interface** statement at the **[edit forwarding-options sampling output]** hierarchy level and specify a monitoring services or adaptive services interface as the output interface. If an output interface is not specified in the sampling configuration, sampling is performed by the Routing Engine.

To specify a flow server in a sampling configuration, include the **flow-server** statement at the **[edit forwarding-options sampling output]** hierarchy level. You must specify the IP address, port number, and flow monitoring version of the destination flow server. Routing Engine-based sampling supports flow

aggregation of up to eight flow servers (version 5 servers and version 8 only) at a time. The export packets are replicated to all flow servers configured to receive them. In contrast, PIC-based sampling allows you to specify just one version 5 flow server and one version 8 server simultaneously. Flow servers operating simultaneously must have different IP addresses.

As part of the output interface statements, you must configure a source address. In contrast, the interface-level statements of **engine-id** and **engine-type** are both added automatically. However, you can override these values with manually configured statements to track different flows with a single flow collector, as needed. When you configure sampling, SNMP input and output interface index information is captured in flow records by default.

```
[edit]
forwarding-options {
    sampling {
        input {
            rate 1;
        }
        family inet {
            output {
                flow-inactive-timeout 15;
                flow-server 10.60.2.1 {
                    port 2055;
                    version 5;
                }
                interface sp-2/0/0 {
                    engine-id 5;
                    engine-type 55;
                    source-address 10.60.2.2;
                }
            }
        }
    }
}
```

Configuring M, MX and T Series Routers for Discard Accounting with a Template

Flow monitoring version 9, which is based on RFC 3954, provides a way to organize flow data into templates. Version 9 also provides a way to actively monitor IPv4, IPv6, MPLS, and peer AS billing traffic. Version 9 is not supported on the AS-I PIC.

To activate templates in flow monitoring, you must configure a template and include that template in the version 9 flow monitoring configuration. Version 9 does not work in conjunction with versions 5 and 8.

To configure a version 9 template, include the **template *template-name*** statement at the [**edit services flow-monitoring version9**] hierarchy level. The Junos OS supports five different templates: **ipv4-template**, **ipv6–template**, **mpls-template**, **mpls-ipv4-template**, and **peer-as-billing-template**. To view the fields selected in each of these templates, see "Flow Monitoring Version 9 Format Output Fields" on page 21.

```
[edit]
services flow-monitoring {
   version9 { # Specifies flow monitoring version 9.
      template mpls { # Specifies template you are configuring.
         template-refresh-rate {
            packets 6000; # The default is 4800 packets and the range is 1–480000
            # packets.
         seconds 90; # The default is 60 seconds and the range is 1–600 seconds.
            option--refresh-rate {
            packets 3000; # The default is 4800 packets and the range is 1–480000
               # packets.
            seconds 30; # The default is 60 seconds and the range is 1–600.
            flow-active-timeout 60; # The default is 60 seconds and the range is
               # 10–600.
            flow-inactive-timeout 30; # The default is 60 seconds and the range 10–600.
            template-refresh-rate seconds 10; # The default is 60 seconds and the
               # range is 10–600
            option-refresh-rate seconds 10; # The default is 60 seconds and the range
               # is 10–600 seconds.
            mpls-template {
               label-positions [1 | 2 | 3]; # Specifies label position for the MPLS template.
            }
         }
      }
   }
}
```

You can export to multiple templates at a time to a maximum of eight flow servers for AS PICs and one flow server for all other PICs. To assign a template to a flow output, include the **template *template-name*** statement at the [**edit forwarding options sampling output flow-server version9**] hierarchy level:

```
[edit]
forwarding-options {
   sampling {
      input {
         family mpls {
```

```
            rate 1;
            run-length 1;
          }
        }
        output {
          flow-server 10.60.2.1 { # The IP address and port of the flow server.
            port 2055;
            source-address 192.0.2.1;
            version9 { # Records are sent to the flow server using version 9 format.
              template { # Indicates a template will organize records.
                mpls; # Records are sent to the MPLS template.
              }
            }
          }
        }
      }
    }
```

Defining a Firewall Filter on M, MX and T Series Routers to Select Traffic for Active Flow Monitoring

The first step in active flow monitoring is to configure the match conditions for acceptable traffic or quarantined traffic. Common match actions for active flow monitoring include **sample**, **discard accounting**, **port-mirror**, and **accept**. To configure, include the desired action statements and a counter as part of the **then** statement in a firewall filter and apply the filter to an interface.

In sampling, the router reviews a portion of the traffic and sends reports about this sample to the flow monitoring server. Discard accounting traffic is counted and monitored, but not forwarded out of the router. Port-mirrored traffic is copied and sent to another interface. Accepted traffic is forwarded to the intended destination.

Most of these match combinations are valid. However, you can either port-mirror or sample with the same traffic at the same time, but not perform more than one action simultaneously on the same packets.

```
[edit]
firewall {
  family inet {
    filter active_filter {
      term quarantined_traffic {
        from {
```

```
                source-address {
                    10.36.1.2/32;
                }
            }
            then {
                count quarantined-counter;
                sample;
                discard accounting;
            }
        }
        term copy_and_forward_the_rest {
            then {
                port-mirror;
                accept;
            }
        }
    }
  }
}
```

Processing IPv4 traffic on an M, MX or T Series Router Using Monitoring services, Adaptive services or Multiservices Interfaces

You configure the monitoring services, adaptive services, or multiservices interfaces with the **family inet** statement so they can process IPv4 traffic. However, you must remember that a monitoring services interface uses an **mo-** prefix and adaptive services and multiservices interfaces use an **sp-** prefix.

```
[edit]
interfaces {
    sp-2/0/0 {
        unit 0 {
            family inet {
                address 10.36.100.1/32 {
                    destination 10.36.100.2;
                }
            }
        }
    }
}
```

Active flow monitoring records leave the router through an export interface to reach the flow monitoring server.

```
[edit]
interfaces {
   fe-1/0/0 {
      unit 0 {
         family inet {
            address 10.60.2.2/30;
         }
      }
   }
}
```

Replicating M, MX and T Series Routing Engine-Based Sampling to Multiple Flow Servers

Routing Engine-based sampling supports up to eight flow servers for both flow monitoring version 5 and version 8 configurations. The total number of flow servers is limited to eight, regardless of how many are configured for version 5 or version 8.

When you configure version 5 or version 8 sampling, the export packets are replicated to all flow servers configured to receive them. If two flow servers are configured to receive version 5 records, both flow servers will receive records for a specified flow.

NOTE: With Routing-Engine-based sampling, if multiple flow servers are configured with version 8 export format, all of them must use the same aggregation type (for example, all flow servers receiving version 8 export could be configured for source-destination aggregation type).

The following configuration example allows replication of export packets to two flow servers.

```
[edit]
forwarding-options {
   sampling {
      input {
         rate 1;
         }
      }
```

```
        output {
            flow-server 10.10.3.2 {
                port 2055;
                version 5;
                source-address 192.168.164.119;
            }
            flow-server 172.17.20.62 {
                port 2055;
                version 5;
                source-address 192.168.164.119;
            }
        }
    }
}
```

Replicating Version 9 Flow Aggregation From M, MX and T Series Routers to Multiple Flow Servers

With this feature, you can configure up to eight flow servers to receive packets for a version 9 flow monitoring template. Once a flow server is configured to receive this data, it will also receive the following periodic version 9 flow monitoring updates:

- Options data
- Template definition

With Routing Engine-based sampling, if multiple collectors are configured with version 8 export format, all of them must use the same aggregation-type.

The option and template definition refresh period is configured on a per-template basis at the [**edit services flow-monitoring**] hierarchy level.

The following configuration example allows replication of version 9 export packets to two flow servers.

```
forwarding-options {
    sampling {
        input {
            family inet {
                rate 1;
            }
        }
        output {
```

```
            flow-server 10.10.3.2 {
                port 2055;
                version9 {
                    template {
                        ipv4;
                    }
                }
            }
            flow-server 172.17.20.62 {
                port 2055;
                version9 {
                    template {
                        ipv4;
                    }
                }
            }
            flow-inactive-timeout 30;
            flow-active-timeout 60;
            interface sp-4/0/0 {
                source-address 10.10.3.4;
            }
        }
    }
}
```

RELATED DOCUMENTATION

- Active Flow Monitoring Overview | 45
- Active Flow Monitoring Overview | 44
- Active Flow Monitoring Applications | 39
- Replicating M, MX and T Series Routing Engine-Based Sampling to Multiple Flow Servers | 83

Configuring Routing Engine-Based Sampling on M, MX and T Series Routers for Export to Multiple Flow Servers

Routing Engine-based sampling supports up to eight flow servers for both version 5 and version 8 configurations. The total number of collectors is limited to eight, regardless of how many are configured for version 5 or version 8. When you configure sampling, the export packets are replicated to all collectors

configured to receive them. If two collectors are configured to receive version 5 records, both collectors will receive records for a specified flow.

The following configuration example allows replication of export packets to two collectors.

```
forwarding-options {
   sampling {
      input {
         family inet {
            rate 1;
         }
      }
      output {
         cflowd 10.10.3.2 {
            port 2055;
            version 5;
            source-address 192.168.164.119;
         }
         cflowd 172.17.20.62 {
            port 2055;
            version 5;
            source-address 192.168.164.119;
         }
      }
   }
}
```

Example: Copying Traffic to a PIC While an M, MX or T Series Router Forwards the Packet to the Original Destination

IN THIS SECTION

Traffic sampling enables you to copy traffic to a Physical Interface Card (PIC) while the router forwards the packet to its original destination. This example describes how to configure a router to perform sampling on the Routing Engine using the **sampled** process. For this method, you configure a filter (input or output) with a matching term that contains the **then sample** statement. In addition, for VPN routing and forwarding (VRF) Routing Engine-based sampling, you configure a VRF routing instance that maps to an interface. Each VRF instance corresponds with a forwarding table. Routes on the interface go into the corresponding forwarding table.

For VRF Routing Engine-based sampling, the kernel queries the correct VRF route table based on the ingress interface index for the received packet. For interfaces configured in VRF, the sampled packets contain the correct input and output interface SNMP index, the source and destination AS numbers, and the source and destination mask.

> **NOTE:** With Junos OS Release 10.1, VRF Routing Engine-based sampling is performed only on IPv4 traffic. You cannot use Routing Engine-based sampling on IPv6 traffic or on MPLS label-switched paths.

This example describes how to configure and verify VRF Routing Engine-based sampling on one router in a four-router topology.

Requirements

This example uses the following hardware and software components:

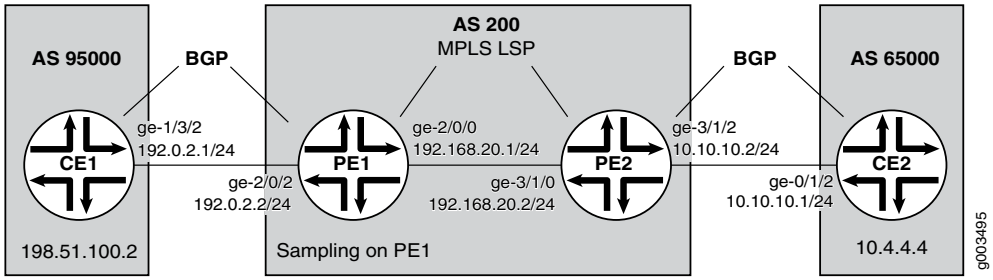
- Junos OS Release 10.1 or later
- M Series, MX Series, or T Series router

Before you configure VRF Routing Engine-Based sampling on your router, be sure you have an active connection between the routers on which you configure sampling. In addition, you need to have an understanding of VRF to configure the interfaces and routing instances that form the basis of the sampling configuration; and an understanding of the BGP, MPLS, and OSPF protocols to configure the other routers in the network to bring up the sampling configuration.

Overview and Topology

The scenario in this example illustrates VRF Routing Engine-based sampling configured on the PE1 router in a four-router network. The CE routers use BGP as the routing protocol to communicate with the PE routers. MPLS LSPs pass traffic between the PE routers. Packets from the CE1 router are sampled on the PE1 router. Regular traffic is forwarded to the original destination (the CE2 router).

Figure 20: Routing Engine-Based Sampling Network Topology



Configuration

IN THIS SECTION

In this configuration example, the VRF Routing Engine-based sampling is configured on the PE1 router that samples the traffic that goes through the interface and routes configured in the VRF. The configurations on the other three routers are included to show the sampling configuration on the PE1 router working in the context of a network.

To configure VRF Routing Engine-based sampling for the network example, perform these tasks:

Configuring the CE1 Router

Step-by-Step Procedure

In this step, you configure interfaces, routing options, protocols, and policy options for the CE1 router. To configure the CE1 router:

1. Configure one interface with two IP addresses. One address is for traffic to the PE1 router; the other address is to check that traffic is flowing to the CE2 router:

   ```
   [edit interfaces]
   user@router-ce1# set ge-1/3/2 unit 0 family inet address 192.0.2.1/24
   user@router-ce1# set ge-1/3/2 unit 0 family inet address 198.51.100.2/8
   ```

2. Configure the autonomous system to establish a connection between BGP peers:

   ```
   [edit routing-options]
   user@router-ce1# set autonomous-system 95000
   ```

3. Configure BGP as the routing protocol between the CE router and the PE router:

   ```
   [edit protocols]
   user@router-ce1# set bgp group to_r1 type external
   user@router-ce1# set bgp group to_r1 export my_lo0_addr
   user@router-ce1# set bgp group to_r1 peer-as 200
   user@router-ce1# set bgp group to_r1 neighbor 192.0.2.2
   ```

4. Configure the policies that ensure that the CE routers exchange routing information. In this example, Router CE1 exchanges routing information with Router CE2:

   ```
   [edit policy-options]
   user@router-ce1# set policy-statement my_lo0_addr term one from protocol direct
   user@router-ce1# set policy-statement my_lo0_addr term one from route-filter 10.255.15.32/32
   exact
   user@router-ce1# set policy-statement my_lo0_addr term one then accept
   user@router-ce1# set policy-statement my_lo0_addr term four from protocol direct
   user@router-ce1# set policy-statement my_lo0_addr term four from route-filter 203.0.113.0/8
   exact
   user@router-ce1# set policy-statement my_lo0_addr term four then accept
   ```

Results

The output below shows the configuration of the CE1 router:

```
[edit]
user@router-ce1# show
```

```
[...Output Truncated...]
interfaces {
    ge-1/3/2 {
        unit 0 {
            family inet {
                address 192.0.2.1/24;
                address 198.51.100.2/8;
            }
        }
    }
}
routing-options {
    autonomous-system 95000;
}
protocols {
    bgp {
        group to_r1 {
            type external;
            export my_lo0_addr;
            peer-as 200;
            neighbor 192.0.2.2;
        }
    }
}
policy-options {
    policy-statement my_lo0_addr {
        term one {
            from {
                protocol direct;
                route-filter 10.255.15.32/32 exact;
            }
            then accept;
        }
        term four {
            from {
                protocol direct;
                route-filter 203.0.113.0/8 exact;
            }
            then accept;
        }
    }
}
```

Configuring the PE1 Router

Step-by-Step Procedure

In this step, you configure a filter with a matching term that contains the **then sample** statement and apply the filter to the ingress interface. You also configure a VRF routing instance with import and export policies. In addition, you configure interfaces, forwarding options, routing options, protocols, and policy options for the PE1 router. To configure the PE1 router:

1. Create the **fw** firewall filter that is applied to the logical interface being sampled:

```
[edit firewall]
user@router-pe1# set family inet filter fw term 1 from protocol tcp
user@router-pe1# set family inet filter fw term 1 from port bgp
user@router-pe1# set family inet filter fw term 1 then accept
user@router-pe1# set family inet filter fw term 2 then sample
```

2. Configure two interfaces, one interface that connects to the CE1 router (**ge-2/0/2**), and another that connects to the PE2 router (**ge-2/0/0**):

```
[edit interfaces]
user@router-pe1# set ge-2/0/2 unit 0 family inet address 192.0.2.2/24
user@router-pe1# set ge-2/0/0 unit 0 family inet address 192.168.20.1/24
user@router-pe1# set ge-2/0/0 unit 0 family mpls
```

3. Enable MPLS on the interface that connects to the PE2 router (**ge-2/0/0**):

```
[edit interfaces]
user@router-pe1# set ge-2/0/0 unit 0 family mpls
```

4. On the interface that connects to the CE1 router, apply the **fw** filter that was configured in the firewall configuration:

```
[edit interfaces]
user@router-pe1# set ge-2/0/2 unit 0 family inet filter input fw
user@router-pe1# set ge-2/0/2 unit 0 family inet filter output fw
```

5. Configure the management (**fxp0**) and loopback (**lo0**) interfaces:

```
[edit interfaces]
user@router-pe1# set fxp0 unit 0 family inet address 192.168.69.153/21
user@router-pe1# set lo0 unit 0 family inet address 127.0.0.1/32
```

6. Configure the **sampled** log file in the **/var/log** directory to record traffic sampling:

```
[edit forwarding-options]
```

```
user@router-pe1# set sampling traceoptions file sampled
user@router-pe1# set sampling traceoptions file world-readable
user@router-pe1# set sampling traceoptions flag all
```

7. Specify the sampling rate and threshold value for traffic sampling:

```
[edit forwarding-options]
user@router-pe1# set sampling input rate 1
user@router-pe1# set sampling input run-length 0
user@router-pe1# set sampling input max-packets-per-second 20000
```

8. Specify active and inactive flow periods, and the router (**198.51.100.2**) that sends out the monitored information:

```
[edit forwarding-options]
user@router-pe1# set sampling family inet output flow-active-timeout 60
user@router-pe1# set sampling family inet output flow-inactive-timeout 60
user@router-pe1# set sampling family inet output flow-server 198.51.100.2 port 2055
user@router-pe1# set sampling family inet output flow-server 198.51.100.2 local-dump
user@router-pe1# set sampling family inet output flow-server 198.51.100.2 version 500
```

9. Configure the autonomous system to establish a connection between BGP peers:

```
[edit routing-options]
user@router-pe1# set autonomous-system 200
```

10. Configure RSVP to support MPLS label-switched paths (LSPs) between the PE routers:

```
[edit protocols]
user@router-pe1# set rsvp interface all
user@router-pe1# set rsvp interface fxp0.0 disable
```

11. Configure an MPLS LSP from the PE1 router to the PE2 router:

```
[edit protocols]
user@router-pe1# set mpls label-switched-path R1toR2 from 192.168.20.1
user@router-pe1# set mpls label-switched-path R1toR2 to 192.168.20.2
user@router-pe1# set mpls interface all
user@router-pe1# set mpls interface fxp0.0 disable
```

12. Configure an internal BGP group for the PE routers. Include the **family inet-vpn unicast** statement to enable BGP to carry network layer reachability information (NLRI) parameters and for BGP peers to only carry unicast routes for forwarding:

```
[edit protocols]
user@router-pe1# set bgp group to_r2 type internal
user@router-pe1# set bgp group to_r2 local-address 192.168.20.1
user@router-pe1# set bgp group to_r2 neighbor 192.168.20.2 family inet-vpn unicast
```

13. Configure OSPF as the interior gateway protocol (IGP) and to compute the MPLS LSPs:

```
user@router-pe1# set ospf traffic-engineering
user@router-pe1# set ospf area 0.0.0.0 interface all
user@router-pe1# set ospf area 0.0.0.0 interface fxp0.0 disable
```

14. Create the extended community that is applied in the policy options configuration:

```
[edit policy-options]
user@router-pe1# set community vpna-comm members target:200:100
```

15. Define the **vpna-export** routing policy that is applied in the **vrf-export** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe1# set policy-statement vpna-export term one from protocol bgp
user@router-pe1# set policy-statement vpna-export term one from protocol direct
user@router-pe1# set policy-statement vpna-export term one then community add vpna-comm
user@router-pe1# set policy-statement vpna-export term one then accept
user@router-pe1# set policy-statement vpna-export term two then reject
```

16. Define the **vpna-import** routing policy that is applied in the **vrf-import** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe1# set policy-statement vpna-import term one from protocol bgp
user@router-pe1# set policy-statement vpna-import term one from community vpna-comm
user@router-pe1# set policy-statement vpna-import term one then accept
user@router-pe1# set policy-statement vpna-import term two then reject
```

17. Configure a VRF routing instance so that routes received from the provider edge-provider edge (PE-PE) session can be imported into any of the instance's VRF secondary routing tables:

```
[edit routing-instances]
user@router-pe1#set vrf1 instance-type vrf set vrf1 interface ge-2/0/2.0
user@router-pe1# set vrf1 route-distinguisher 10.255.15.51:1
user@router-pe1# set vrf1 vrf-import vpna-import
user@router-pe1# set vrf1 vrf-export vpna-export
```

```
user@router-pe1# set vrf1 protocols bgp group customer type external
user@router-pe1# set vrf1 protocols bgp group customer peer-as 95000
user@router-pe1# set vrf1 protocols bgp group customer as-override
user@router-pe1# set vrf1 protocols bgp group customer neighbor 192.168.30.1
user@router-pe1# set vrf1 protocols bgp group customer neighbor 192.0.2.1
```

Results

Check the results of the configuration for the PE1 router:

```
user@router-pe1> show configuration
[...Output Truncated...]
}
interfaces {
    ge-2/0/0 {
        unit 0 {
            family inet {
                address 192.168.20.1/24;
            }
            family mpls;
        }
    }
    ge-2/0/2 {
        unit 0 {
            family inet {
                filter {
                    input fw;
                    output fw;
                }
                address 192.0.2.2/24;
            }
        }
    }
    fxp0 {
        unit 0 {
            family inet {
                address 192.168.69.153/21;
            }
        }
    }
    lo0 {
        unit 0 {
            family inet {
```

```
                address 127.0.0.1/32;
            }
        }
    }
}
forwarding-options {
    sampling {
        traceoptions {
            file sampled world-readable;
            flag all;
        }
        input {
            rate 1;
            run-length 0;
            max-packets-per-second 20000;
        }
        family inet {
            output {
                flow-inactive-timeout 60;
                flow-active-timeout 60;
                flow-server 198.51.100.2 {
                    port 2055;
                    local-dump;
                    version 500;
                }
            }
        }
    }
}
routing-options {
[...Output Truncated...]
    autonomous-system 200;
}
protocols {
    rsvp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        label-switched-path R1toR2 {
            from 192.168.20.1;
            to 192.168.20.2;
```

```
        }
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group to_r2 {
            type internal;
            local-address 192.168.20.1;
            neighbor 192.168.20.2 {
                family inet-vpn {
                    unicast;
                }
            }
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
}
policy-options {
    policy-statement vpna-export {
        term one {
            from protocol [ bgp direct ];
            then {
                community add vpna-comm;
                accept;
            }
        }
        term two {
            then reject;
        }
    }
    policy-statement vpna-import {
        term one {
            from {
                protocol bgp;
```

```
                community vpna-comm;
            }
            then accept;
        }
        term two {
            then reject;
        }
    }
    community vpna-comm members target:200:100;
}
firewall {
    family inet {
        filter fw {
            term 1 {
                from {
                    protocol tcp;
                    port bgp;
                }
                then accept;
            }
            term 2 {
                then sample;
            }
        }
    }
}
routing-instances {
    vrf1 {
        instance-type vrf;
        interface ge-2/0/2.0;
        route-distinguisher 10.255.15.51:1;
        vrf-import vpna-import;
        vrf-export vpna-export;
        protocols {
            bgp {
                group customer {
                    type external;
                    peer-as 95000;
                    as-override;
                    neighbor 192.168.30.1;
                    neighbor 192.0.2.1;
                }
            }
        }
```

```
    }
}
```

Configuring the PE2 Router

Step-by-Step Procedure

In this step, you configure a filter with a matching term that contains the **then sample** statement and apply the filter to the ingress interface. You also configure a VRF routing instance with import and export policies. In addition, you configure interfaces, forwarding options, routing options, protocols, and policy options for the PE2 router. To configure the PE2 router:

1. Create the **fw** firewall filter that is applied to the logical interface being sampled:

   ```
   [edit firewall]
   user@router-pe2# set family inet filter fw term 1 from protocol tcp
   user@router-pe2# set family inet filter fw term 1 from port bgp
   user@router-pe2# set family inet filter fw term 1 then accept
   user@router-pe2# set family inet filter fw term 2 then sample
   user@router-pe2# set family inet filter fw term 2 then accept
   ```

2. Configure two interfaces, one interface that connects to the CE2 router (**ge-3/1/2**), and another that connects to the PE1 router (**ge-3/1/0**):

   ```
   [edit interfaces]
   user@router-pe2# set ge-3/1/0 unit 0 family inet address 192.168.20.2/24
   user@router-pe2#  set ge-3/1/0 unit 0 family mpls
   user@router-pe2#  set ge-3/1/2 unit 0 family inet address 10.10.10.2/24
   ```

3. Enable MPLS on the interface that connects to the PE1 router (**ge-3/1/0**):

   ```
   [edit interfaces]
   user@router-pe2#  set ge-3/1/0 unit 0 family mpls
   ```

4. On the interface that connects to the CE2 router, apply the **fw** filter that was configured in the firewall configuration:

   ```
   [edit interfaces]
   user@router-pe2#  set ge-3/1/2 unit 0 family inet filter input fw
   user@router-pe2#  set ge-3/1/2 unit 0 family inet filter output fw
   ```

5. Configure the **sampled** log file in the **/var/log** directory to record traffic sampling:

   ```
   [edit forwarding-options]
   ```

```
user@router-pe2# set sampling traceoptions file sampled
user@router-pe2# set sampling traceoptions file world-readable
user@router-pe1# set sampling traceoptions flag all
```

6. Specify the sampling rate and threshold value for traffic sampling:

```
[edit forwarding-options]
user@router-pe2# set sampling input rate 1
user@router-pe2# set sampling input run-length 0
user@router-pe2# set sampling input max-packets-per-second 20000
```

7. Specify active and inactive flow periods, and the router (**198.51.100.2**) that sends out the monitored information:

```
[edit forwarding-options]
user@router-pe2# set sampling family inet output flow-active-timeout 60
user@router-pe2# set sampling family inet output flow-inactive-timeout 60
user@router-pe2# set sampling family inet output flow-server 198.51.100.2 port 2055
user@router-pe2# set sampling family inet output flow-server 198.51.100.2 local-dump
user@router-pe2# set sampling family inet output flow-server 198.51.100.2 version 500
```

8. Configure the autonomous system to establish a connection between BGP peers:

```
[edit routing-options]
user@router-pe2# set autonomous-system 200
```

9. Configure RSVP to support MPLS label-switched paths (LSPs) between the PE routers:

```
[edit protocols]
user@router-pe2# set rsvp interface all
user@router-pe2# set rsvp interface fxp0.0 disable
```

10. Configure an MPLS LSP from the PE2 router to the PE1 router:

```
[edit protocols]
user@router-pe2# set mpls label-switched-path R2toR1 from 192.168.20.2
user@router-pe2# set mpls label-switched-path R2toR1 to 192.168.20.1
user@router-pe2# set mpls interface all
user@router-pe2# set mpls interface fxp0.0 disable
```

11. Configure an internal BGP group for the PE routers. Include the **family inet-vpn unicast** statement to enable BGP to carry network layer reachability information (NLRI) parameters and for BGP peers to only carry unicast routes for forwarding:

```
[edit protocols]
user@router-pe2# set bgp group to_r1 type internal
user@router-pe2# set bgp group to_r1 local-address 192.168.20.2
user@router-pe2# set bgp group to_r1 neighbor 192.168.20.1 family inet-vpn unicast
```

12. Configure OSPF as the interior gateway protocol (IGP) and to compute the MPLS LSPs:

```
[edit protocols]
user@router-pe2# set ospf traffic-engineering
user@router-pe2# set ospf area 0.0.0.0 interface all
user@router-pe2# set ospf area 0.0.0.0 interface fxp0.0 disable
```

13. Create the extended community that is applied in the policy options configuration:

```
[edit policy-options]
user@router-pe2# set community vpna-comm members target:200:100
```

14. Define the **vpna-export** routing policy that is applied in the **vrf-export** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe2# set policy-statement vpna-export term one from protocol bgp
user@router-pe2# set policy-statement vpna-export term one from protocol direct
user@router-pe2# set policy-statement vpna-export term one then community add vpna-comm
user@router-pe2# set policy-statement vpna-export term one then accept
user@router-pe2# set policy-statement vpna-export term two then reject
```

15. Define the **vpna-import** routing policy that is applied in the **vrf-import** statement in the routing instance configuration. Also, apply the **vpna-comm** community from which routes are learned:

```
[edit policy-options]
user@router-pe2# set policy-statement vpna-import term one from protocol bgp
user@router-pe2# set policy-statement vpna-import term one from community vpna-comm
user@router-pe2# set policy-statement vpna-import term one then accept
user@router-pe2# set policy-statement vpna-import term two then reject
```

16. Configure a VRF routing instance so that routes received from the provider edge-provider edge (PE-PE) session can be imported into any of the instance's VRF secondary routing tables:

```
[edit routing-instances]
user@router-pe2# set vrf1 instance-type vrf
user@router-pe2# set vrf1 interface ge-3/1/2.0
user@router-pe2# set vrf1 route-distinguisher 10.255.19.12:1
```

```
user@router-pe2# set vrf1 vrf-import vpna-import
user@router-pe2# set vrf1 vrf-export vpna-export
user@router-pe2#set vrf1 protocols bgp group R3-R4 type external
user@router-pe2# set vrf1 protocols bgp group R3-R4 peer-as 65000
user@router-pe2# set vrf1 protocols bgp group R3-R4 as-override
user@router-pe2# set vrf1 protocols bgp group R3-R4 neighbor 10.10.10.1
```

Results

Check the results of the configuration for the PE2 router:

```
user@router-pe2> show configuration
[...Output Truncated...]
}
interfaces {
    ge-3/1/0 {
        unit 0 {
            family inet {
                address 192.168.20.2/24;
            }
            family mpls;
        }
    }
    ge-3/1/2 {
        unit 0 {
            family inet {
                filter {
                    input fw;
                    output fw;
                }
                address 10.10.10.2/24;
            }
        }
    }
}
forwarding-options {
    sampling {
        traceoptions {
            file sampled world-readable;
            flag all;
        }
        input {
            rate 1;
```

```
            run-length 0;
            max-packets-per-second 20000;
        }
        family inet {
            output {
                flow-inactive-timeout 60;
                flow-active-timeout 60;
                flow-server 198.51.100.2 {
                    port 2055;
                    local-dump;
                    version 500;
                }
            }
        }
    }
}
routing-options {
[...Output Truncated...]
    autonomous-system 200;
}
protocols {
    rsvp {
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    mpls {
        label-switched-path R2toR1 {
            from 192.168.20.2;
            to 192.168.20.1;
        }
        interface all;
        interface fxp0.0 {
            disable;
        }
    }
    bgp {
        group to_r1 {
            type internal;
            local-address 192.168.20.2;
            neighbor 192.168.20.1 {
                family inet-vpn {
                    unicast;
```

```
                }
            }
            neighbor 192.0.2.1;
        }
    }
    ospf {
        traffic-engineering;
        area 0.0.0.0 {
            interface all;
            interface fxp0.0 {
                disable;
            }
        }
    }
}
policy-options {
    policy-statement vpna-export {
        term one {
            from protocol [ bgp direct ];
            then {
                community add vpna-comm;
                accept;
            }
        }
        term two {
            then reject;
        }
    }
    policy-statement vpna-import {
        term one {
            from {
                protocol bgp;
                community vpna-comm;
            }
            then accept;
        }
        term two {
            then reject;
        }
    }
    community vpna-comm members target:200:100;
}
firewall {
    family inet {
```

```
        filter fw {
            term 1 {
                from {
                    protocol tcp;
                    port bgp;
                }
                then accept;
            }
            term 2 {
                then {
                    sample;
                    accept;
                }
            }
        }
    }
}
routing-instances {
    vrf1 {
        instance-type vrf;
        interface ge-3/1/2.0;
        route-distinguisher 10.255.19.12:1;
        vrf-import vpna-import;
        vrf-export vpna-export;
        protocols {
            bgp {
                group R3-R4 {
                    type external;
                    peer-as 65000;
                    as-override;
                    neighbor 10.10.10.1;
                }
            }
        }
    }
}
```

Configuring the CE2 Router

Step-by-Step Procedure

In this step, you configure interfaces, routing options, protocols, and policy options for the CE2 router. To configure the CE2 router:

1. Configure one interface with two IP addresses. One address is for traffic to the PE2 router and the other address is to check that traffic is flowing from the CE1 router:

   ```
   [edit interfaces]
   user@router-ce2# set ge-0/1/2 unit 0 family inet address 10.10.10.1/24
   user@router-ce2#  set ge-0/1/2 unit 0 family inet address 10.4.4.4/16
   ```

2. Configure the autonomous system to establish a connection between BGP peers:

   ```
   [edit routing-options]
   user@router-ce1# set autonomous-system 65000
   ```

3. Configure BGP as the routing protocol between the CE and the PE routers:

   ```
   [edit protocols]
   user@router-ce2# set bgp group R3-R4 type external
   user@router-ce2#  set bgp group R3-R4 export l3vpn-policy
   user@router-ce2#  set bgp group R3-R4 peer-as 200
   user@router-ce2#  set bgp group R3-R4 neighbor 10.10.10.2
   ```

4. Configure the policies that ensure that the CE routers exchange routing information. In this example, Router CE2 exchanges routing information with Router CE1:

   ```
   [edit policy-options]
   user@router-ce2# set policy-statement l3vpn-policy term one from protocol direct
   user@router-ce2#  set policy-statement l3vpn-policy term one from route-filter 10.255.15.75/32
   exact
   user@router-ce2# set policy-statement l3vpn-policy term one then accept
   user@router-ce2# set policy-statement l3vpn-policy term two from protocol direct
   user@router-ce2# set policy-statement l3vpn-policy term two from route-filter 10.4.0.0/16 exact
   user@router-ce2#  set policy-statement l3vpn-policy term two then accept
   ```

Results

The output below shows the configuration of the CE2 router:

```
[edit]
user@router-ce2# show
[...Output Truncated...]
```

```
interfaces {
    ge-0/1/2 {
        unit 0 {
            family inet {
                address 10.10.10.1/24;
                address 10.4.4.4/16;
            }
        }
    }
}
routing-options {
    autonomous-system 65000;
}
protocols {
    bgp {
        group R3-R4 {
            type external;
            export l3vpn-policy;
            peer-as 200;
            neighbor 10.10.10.2;
        }
    }
}
policy-options {
    policy-statement l3vpn-policy {
        term one {
            from {
                protocol direct;
                route-filter 10.255.15.75/32 exact;
            }
            then accept;
        }
        term two {
            from {
                protocol direct;
                route-filter 10.4.0.0/16 exact;
            }
            then accept;
        }
    }
}
```

Verification

IN THIS SECTION

After you have completed the configuration of the four routers, you can verify that traffic is flowing from the CE1 router to the CE2 router, and you can observe the sampled traffic from two locations. To confirm that the configuration is working properly, perform these tasks:

Verifying the Traffic Flow Between the CE Routers

Purpose

Use the **ping** command to verify traffic between the CE routers.

Action

From the CE1 router, issue the **ping** command to the CE2 router:

```
user@router-ce2> ping 10.4.4.4 source 198.51.100.2
PING 10.4.4.4 (10.4.4.4): 56 data bytes
64 bytes from 10.4.4.4: icmp_seq=0 ttl=64 time=0.861 ms
64 bytes from 10.4.4.4: icmp_seq=1 ttl=64 time=0.869 ms
64 bytes from 10.4.4.4: icmp_seq=2 ttl=64 time=0.786 ms
^C
--- 10.4.4.4 ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.786/0.839/0.869/0.037 ms
```

Meaning

The output from the **ping** command shows that the **ping** command was successful. Traffic is flowing between the CE routers.

Verifying Sampled Traffic

Purpose

You can observe the sampled traffic using the **show log sampled** command from the CLI or from the router shell using the **tail –f /var/log/sampled** command. In addition, you can collect the logs in a flowcollector. The same information appears in the output of both commands and in the flow collector. For information about using a flow collector, see "“Sending cflowd Records to Flow Collector Interfaces” on page 282" and "“Example: Configuring a Flow Collector Interface on an M, MX or T Series Router” on page 253."

Action

From the PE1 router, use the **show log sampled** command:

```
user@router-pe1> show log sampled
[...Output Truncated...]
Nov 16 23:24:19    Src addr: 198.51.100.2
Nov 16 23:24:19    Dst addr: 10.4.4.4
Nov 16 23:24:19    Nhop addr: 192.168.20.2
Nov 16 23:24:19   Input interface: 503      # SNMP index of the incoming interface on PE1
Nov 16 23:24:19   Output interface: 505      # SNMP index of the outgoing interface on PE1
Nov 16 23:24:19    Pkts in flow: 5
Nov 16 23:24:19    Bytes in flow: 420
Nov 16 23:24:19    Start time of flow: 602411369
Nov 16 23:24:19    End time of flow: 602415369
Nov 16 23:24:19    Src port: 0
Nov 16 23:24:19    Dst port: 2048
Nov 16 23:24:19    TCP flags: 0x0
Nov 16 23:24:19    IP proto num: 1
Nov 16 23:24:19    TOS: 0x0
Nov 16 23:24:19   Src AS: 95000      # The autonomous system of CE1
Nov 16 23:24:19   Dst AS: 65000,,,,,# The autonomous system of CE2
Nov 16 23:24:19   Src netmask len: 8
Nov 16 23:24:19   Dst netmask len: 16
Nov 16 23:24:19 cflowd header:
Nov 16 23:24:19   Num-records: 1
Nov 16 23:24:19   Version: 500
Nov 16 23:24:19   Flow seq num: 13
Nov 16 23:24:19   Sys Uptime: 602450382 (msecs)
Nov 16 23:24:19   Time-since-epoch: 1258413859 (secs)
Nov 16 23:24:19   Engine id: 0
Nov 16 23:24:19   Engine type: 0
Nov 16 23:24:19   Sample interval: 1
[...Output Truncated...]
```

Meaning

The output from the **show log sampled** command shows the correct SNMP index for the incoming and outgoing interfaces on the PE1 router. Also, the source and destination addresses for the autonomous systems for the two CE routers are correct.

Cross Verifying Sampled Traffic

Purpose

You can also double check that the sampled traffic is the correct traffic by using the **show interface** ***interface-name-fpc/pic/port.unit-number*** **| match SNMP** command and the **show route** ***route-name*** **detail** command.

Action

The following output is a cross check of the output in the "Verifying Sampled Traffic" on page 107 task:

```
user@router-pe1> show interfaces ge-2/0/2.0 | match SNMP
  Logical interface ge-2/0/2.0 (Index 76) (SNMP ifIndex 503)
    Flags: SNMP-Traps 0x4000000 Encapsulation: ENET2
```

```
user@router-pe1> show route 10.4.4.4 detail

vrf1.inet.0: 5 destinations, 5 routes (5 active, 0 holddown, 0 hidden)
10.4.0.0/16 (1 entry, 1 announced)
        *BGP    Preference: 170/-101
                Route Distinguisher: 10.255.19.12:1
                Next hop type: Indirect
                Next-hop reference count: 6
                Source: 192.168.20.2
                Next hop type: Router, Next hop index: 659
                Next hop: 192.168.20.2 via ge-2/0/0.0 weight 0x1, selected
                Label operation: Push 299776
                Protocol next hop: 192.168.20.2
                Push 299776
                Indirect next hop: 8e6f780 1048574
                State: <Secondary Active Int Ext>
                Local AS:   200 Peer AS:   200
                Age: 3d 19:49:32  Metric2: 65535
                Task: BGP_200.20.20.20.2+179
                Announcement bits (3): 0-RT 1-BGP RT Background 2-KRT
                AS path: 65000 I
                AS path: Recorded
                Communities: target:200:100
                Import Accepted
```

```
                VPN Label: 299776
                Localpref: 100
                Router ID: 10.10.10.2
                Primary Routing Table bgp.l3vpn.0
```

Meaning

The output of the **show interfaces ge-2/0/2.0 | match SNMP** command shows that the SNMP ifIndex field has the same value (**503**) as the output for the **show log sampled** command in the "Verifying Sampled Traffic" on page 107 task, indicating that the intended traffic is being sampled.

The output of the **show route 10.4.4.4 detail** command shows that the source address **10.4.4.4**, the source mask (**16**), and the source AS (**65000**) have the same values as the output for the **show log sampled** command in the "Verifying Sampled Traffic" on page 107 task, indicating that the intended traffic is being sampled.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

Configuring an Aggregate Export Timer on M, MX and T Series Routers for Version 8 Records

When you use flow monitoring version 8 records for active flow monitoring, you can configure an aggregate export timer. To configure this timer, include the **aggregate-export-interval** statement at the **[edit forwarding-options sampling output]** hierarchy level. The timer value has a default minimum setting of 90 seconds and a maximum value of 1800 seconds.

```
[edit]
forwarding-options {
    sampling {
        output {
            aggregate-export-interval duration;
        }
    }
}
```

Rerouting Packets on an M, MX or T Series Router with Port Mirroring

You can copy packets and reroute them to another interface by using port mirroring. To send packet copies to an interface, include the **interface** statement at the **[edit forwarding-options port-mirroring** family ***family-name*** **output]** hierarchy level and specify the interface to receive the traffic.

You can even send port-mirrored traffic to a monitoring services or adaptive services interface. If you choose this option, accepted traffic is copied and the packet copies are sent to the services interface for flow processing.

To configure how often packets are copied from the monitored traffic, include the **rate** statement at the **[edit forwarding-options port-mirroring** family ***family-name*** **input]** hierarchy level. A rate of **1** port-mirrors every packet, while a rate of **10** port-mirrors every tenth packet.

```
[edit]
forwarding-options {
    port-mirroring {
        family (inet | inet6) {
            input {
                rate 1;
            }
            output {
                interface sp-2/0/0.0;
            }
        }
    }
}
```

Option: Configuring Port Mirroring with Filter-Based Forwarding and a Monitoring Group

For active flow monitoring, you can load-balance traffic across multiple Monitoring Services PICs using the same method as passive flow monitoring. The only difference is that you do not configure the input interface with the **passive-monitor-mode** statement at the **[edit interfaces *interface-name*]** hierarchy level.

To load-balance traffic for active flow monitoring, port-mirror the incoming packets to a tunnel services interface. Redirect this copy of the traffic to a filter-based forwarding instance by applying a firewall filter to the tunnel services interface. Configure the instance to send the traffic to a group of monitoring services interfaces. Finally, use a monitoring group to send flow records from the monitoring services interfaces to a flow server.

NOTE: When you load-balance port-mirrored traffic across several Monitoring Services interfaces, there are some limitations:

- The original Monitoring Services PIC supports this method. You cannot use a Monitoring Services II PIC.
- You must use the suite of **show passive-monitoring** commands to monitor traffic. The **show services accounting** commands are not supported.
- Because load-balanced traffic is routed through the Tunnel Services PIC, the total throughput of the load-balanced traffic coming from the Monitoring Services PICs cannot exceed the bandwidth of the tunnel interface.

For detailed information on this method, see "Copying and Redirecting Traffic with Port Mirroring and Filter-Based Forwarding" on page 229.

Sending Port-Mirrored Traffic from an M, MX or T Series Router to Multiple Export Interfaces by Using Next-Hop Groups

To send port-mirrored traffic to multiple flow servers or packet analyzers, you can use the **next-hop-group** statement. The router can make up to 16 copies of traffic per group and send the traffic to the next-hop group members you configure. 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 configure multiple port mirroring with next-hop groups, include the **next-hop-group** statement at the **[edit forwarding-options]** hierarchy level.

You must port-mirror the initial traffic to a tunnel interface so that it can be filtered and duplicated. Also, you need configure only the interface names for point-to-point interfaces, but you must configure the interface names and a next hop for multipoint interfaces (such as Ethernet).

```
[edit]
forwarding-options {
   port-mirroring {
      family inet {
         input {
            rate 1;
         }
         output {
            interface vt-3/3/0.1;
            no-filter-check;
```

```
            }
        }
    }
    next-hop-group ftp-traffic {
        interface so-4/3/0.0;
        interface so-0/3/0.0;
    }
    next-hop-group http-traffic {
        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;
    }
}
```

NOTE: **Next-hop groups** are supported on M Series routers only, except the M120 router and the M320 router.

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 21: Active Flow Monitoring—Multiple Port Mirroring with Next-Hop Groups Topology Diagram

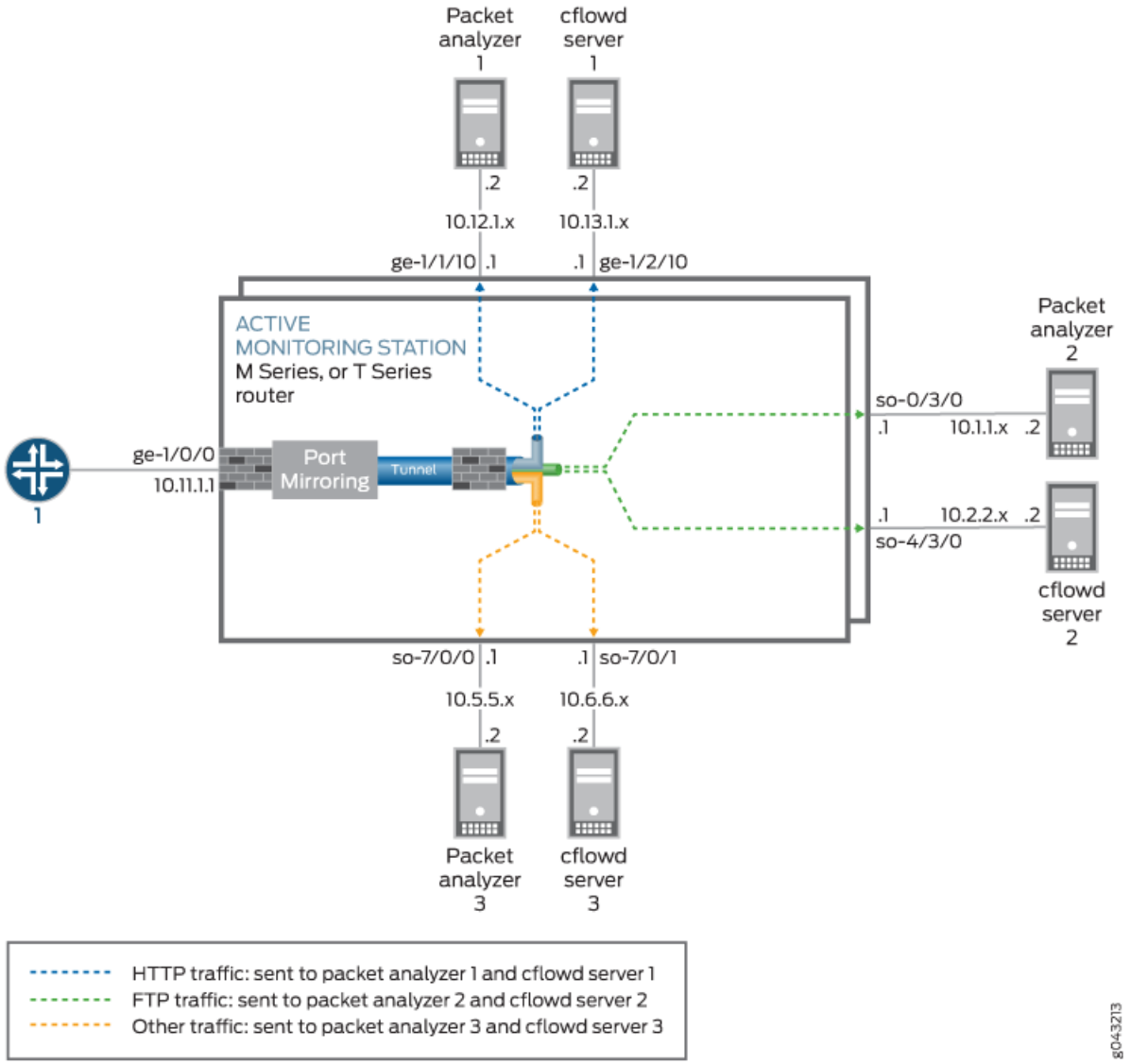


Figure 21 on page 173 shows an example of how to configure multiple port mirroring with next-hop groups. All traffic enters the monitoring router at interface ge-1/0/0. A firewall filter counts and port-mirrors all incoming packets to a Tunnel Services PIC. A second filter is applied to the tunnel interface 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;
            }
        }
    }
```

```
}
```

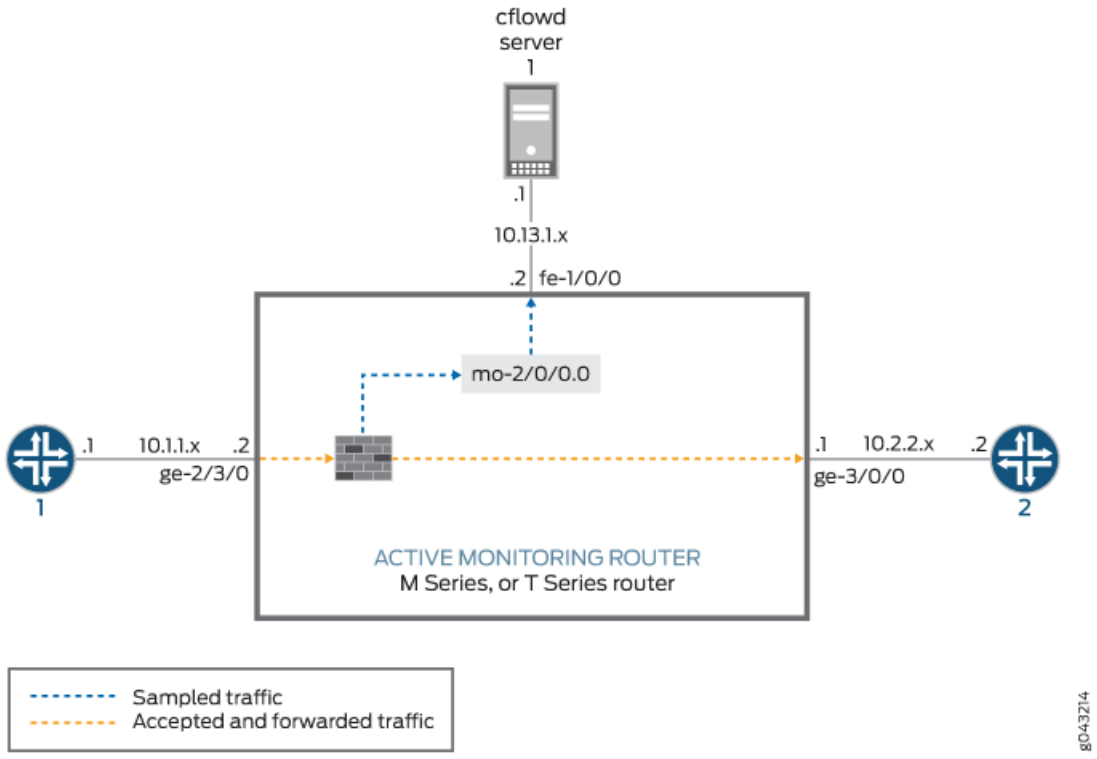
RELATED DOCUMENTATION

Understanding Port Mirroring | 546

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

Example: Sampling Configuration for M, MX and T Series Routers

Figure 22: Active Flow Monitoring—Sampling Configuration Topology Diagram



In Figure 17 on page 113, traffic from Router 1 arrives on the monitoring router's Gigabit Ethernet **ge-2/3/0** interface. The exit interface on the monitoring router that leads to destination Router 2 is **ge-3/0/0**. In active flow monitoring, both the input interface and exit interface can be any interface type (such as SONET/SDH, Gigabit Ethernet, and so on). The export interface leading to the flow server is **fe-1/0/0**.

Configure a firewall filter to sample, count, and accept all traffic. Apply the filter to the input interface, and configure the exit interface (for traffic forwarding), the adaptive services interface (for flow processing), and the export interface (for exporting flow records).

Configure sampling at the **[edit forwarding-options]** hierarchy level. Include the IP address and port of the flow server with the **flow-server** statement and specify the adaptive services interface to be used for flow record processing with the **interface** statement at the **[edit forwarding-options sampling]** hierarchy level.

Router 1

```
[edit]
interfaces {
    sp-2/0/0 { # This adaptive services interface creates the flow records.
        unit 0 {
            family inet {
                address 10.5.5.1/32 {
                    destination 10.5.5.2;
                }
            }
        }
    }
    fe-1/0/0 { # This is the interface where records are sent to the flow server.
        unit 0 {
            family inet {
                address 10.60.2.2/30;
            }
        }
    }
    ge-2/3/0 { # This is the input interface where all traffic enters the router.
        unit 0 {
            family inet {
                filter {
                    input catch_all; # This is where the firewall filter is applied.
                }
                address 10.1.1.1/20;
            }
        }
    }
    ge-3/0/0 { # This is the interface where the original traffic is forwarded.
        unit 0 {
            family inet {
                address 10.2.2.1/24;
            }
```

```
        }
    }
}
forwarding-options {
    sampling { # Traffic is sampled and sent to a flow server.
        input {
            rate 1; # Samples 1 out of  x  packets (here, a rate of 1 sample per packet).
            }
        }
        family inet {
            output {
                flow-server 10.60.2.1 { # The IP address and port of the flow server.
                    port 2055;
                    version 5; # Records are sent to the flow server using version 5 format.
                }
                flow-inactive-timeout 15;
                flow-active-timeout 60;
                interface sp-2/0/0 { # Adding an interface here enables PIC-based sampling.
                    engine-id 5; # Engine statements are dynamic, but can be configured.
                    engine-type 55;
                    source-address 10.60.2.2; # You must configure this statement.
                }
            }
    }
}
firewall {
    family inet {
        filter catch_all { # Apply this filter on the input interface.
            term default {
                then {
                    sample;
                    count counter1;
                    accept;
                }
            }
        }
    }
}
```

Verifying Your Work

To verify that your configuration is correct, use the following commands on the monitoring station that is configured for active flow monitoring:

- **show services accounting errors**
- **show services accounting (flow | flow-detail)**
- **show services accounting memory**
- **show services accounting packet-size-distribution**
- **show services accounting status**
- **show services accounting usage**
- **show services accounting aggregation template template-name *name* (detail | extensive | terse)** (version 9 only)

Most active flow monitoring operational mode commands contain equivalent output information to the following passive flow monitoring commands:

- **show services accounting errors = show passive-monitoring error**
- **show services accounting flow = show passive-monitoring flow**
- **show services accounting memory = show passive-monitoring memory**
- **show services accounting status = show passive-monitoring status**
- **show services accounting usage = show passive-monitoring usage**

The active flow monitoring commands can be used with most active flow monitoring applications, including sampling, discard accounting, port mirroring, and multiple port mirroring. However, you can use the passive flow monitoring commands only with configurations that contain a monitoring group at the **[edit forwarding-options monitoring]** hierarchy level.

The following shows the output of the **show** commands used with the configuration example:

`user@router1>` **show services accounting errors**

```
Service Accounting interface: sp-2/0/0, Local interface index: 542
Service name: (default sampling)
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory overload: No, PPS overload: No, BPS overload: Yes
```

```
user@router1> show services accounting flow-detail limit 10
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
Protocol   Source         Source  Destination Destination    Packet        Byte
           Address          Port  Address            Port     count       count
udp(17)    10.1.1.2           53  10.0.0.1             53      4329     3386035
ip(0)      10.1.1.2            0  10.0.0.2              0      4785     3719654
ip(0)      10.1.1.2            0  10.0.1.2              0      4530     3518769
udp(17)    10.1.1.2            0  10.0.7.1              0      5011     3916767
tcp(6)     10.1.1.2           20  10.3.0.1             20         1        1494
tcp(6)     10.1.1.2           20  10.168.80.1          20         1         677
tcp(6)     10.1.1.2           20  10.69.192.1          20         1         446
tcp(6)     10.1.1.2           20  10.239.240.1         20         1        1426
tcp(6)     10.1.1.2           20  10.126.160.1         20         1         889
tcp(6)     10.1.1.2           20  10.71.224.1          20         1        1046

user@router1> show services accounting memory
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
  Memory utilization
    Allocation count: 437340, Free count: 430681, Maximum allocated: 6782
    Allocations per second: 3366, Frees per second: 6412
   Total memory used (in bytes): 133416928, Total memory free (in bytes): 133961744

user@router1> show services accounting packet-size-distribution
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
Range start     Range end     Number of packets     Percentage packets
         64            96               1705156                    100

user@router1> show services accounting status
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
  Interface state: Monitoring
  Group index: 0
  Export interval: 60 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 55, Engine ID: 5
  Route record count: 13, IFL to SNMP index count: 30, AS count: 1
  Time set: Yes, Configuration set: Yes
  Route record set: Yes, IFL SNMP map set: Yes

user@router1> show services accounting usage
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: (default sampling)
```

```
CPU utilization
  Uptime: 4790345 milliseconds, Interrupt time: 1668537848 microseconds
  Load (5 second): 71%, Load (1 minute): 63%
```

Example: Sampling Instance Configuration

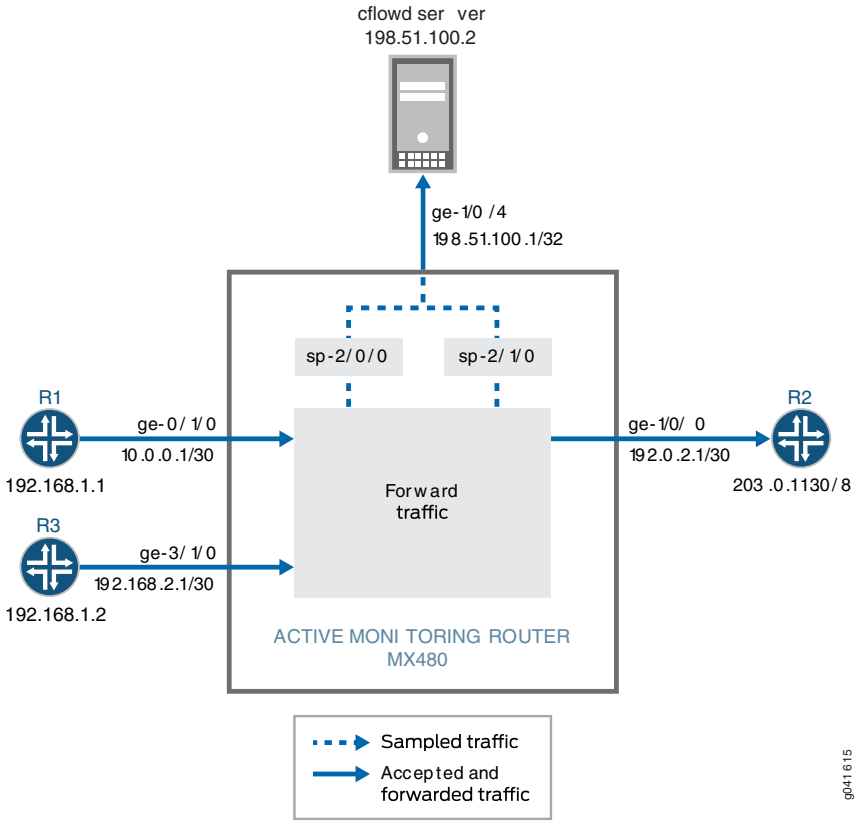
IN THIS SECTION

You can configure active sampling using a sampling instance and associate that sampling instance to a particular FPC, MPC, or DPC. In addition, you can define multiple sampling instances associated with multiple destinations and protocol families per sampling instance destination.

Example Network Details

The following example shows the configuration of two sampling instances on an MX480 router running Junos OS Release 9.6.

Figure 23: Active Flow Monitoring—Sampling Instance Configuration Topology Diagram



In Figure 18 on page 119, packets from Router 1 arrive on the monitoring router's Gigabit Ethernet **ge-0/1/0** interface, the packets are sampled by the services interface **sp-2/0/0** and sent to the cflowd server by the export interface **ge-1/0/4**. Packets from Router 3 arrive on the monitoring router's Gigabit Ethernet **ge-3/1/0** interface, the packets are sampled by the services interface **sp-2/1/0** and sent to the cflowd server by the export interface **ge-1/0/4**. Normal traffic flow from **ge-0/1/0** and **ge-3/1/0** to **ge-1/0/0** and on to Router 2 continues undisturbed during the sampling process. In active flow monitoring, both the input interface and exit interface can be any interface type (such as SONET/SDH, Gigabit Ethernet, and so on).

Only one sampling instance can be attached to an FPC, MPC, or DPC. Multiple families can be configured under a sampling instance. Each family can have its own collector address. You can define sampling instances and attach each instance to different FPCs, or a single sampling instance can be attached to all FPCs.

The sampling configuration for this example includes the following:

- Two sampling instances, **s0** and **s1**, configured to collect sampling data at the **[edit forwarding-options]** hierarchy level. The **flow-server** statement includes the IP address, port, and template of the flow server. The **interface** statement includes the services interface, **sp-2/0/0** or **sp-2/1/0**, for flow record processing, and the source address of the incoming router on the sampled interface.

- The binding of the two sampling instances to FPCs 0 and 3. These are configured with the **sampling-instance** statement at the **[edit chassis fpc *slot*]** hierarchy level.
- Sampling activated on the input interfaces **ge-0/1/0** and **ge-3/1/0** using the **sampling** statement at the **[edit interfaces *interface-name* unit *unit-number* family *family*]** hierarchy level.

In this example, the **ping** command is issued on Router 1 to Router 2 via the MX480 router to generate traffic. After the packets are generated, **show** commands are issued to verify that the sampling configuration is working as expected.

Example Router Configuration

The following output shows the configuration of an MX480 router with two sampling instances.

```
user@MX480-router> show configuration
[...Output Truncated...]
}
chassis {
    fpc 0 { # The fpc number is associated with the interface on which sampling
is enabled, ge-0/1/0 in this statement.
        sampling-instance s0;
    }
    fpc 3 { # The fpc number is associated with the interface on which sampling
is enabled, ge-3/1/0 in this statement.
        sampling-instance s1;
    }
}
interfaces {
    ge-0/1/0 { # This interface has sampling activated.
        unit 0 {
            family inet {
                sampling { # Here sampling is activated.
                    input;
                }
                address 10.0.0.1/30;
            }
        }
    }
    ge-1/0/0 { # The interface on which packets are exiting the router.
        unit 0 {
            family inet {
                address 192.0.2.1/30;
            }
        }
    }
```

```
    ge-1/0/4 { # The interface connected to the cflowd server.
        unit 0 {
            family inet {
                address 198.51.100.1/32;
            }
        }
    }
    sp-2/0/0 { # The service interface that samples the packets from Router 1.
        unit 0 {
            family inet;
        }
    }
    sp-2/1/0 { # The service interface that samples the packets from Router 3.
        unit 0 {
            family inet;
        }
    }
    ge-3/1/0 { # This interface has sampling activated.
        unit 0 {
            family inet {
                sampling { # Here sampling is activated.
                    input;
                }
                address 192.168.2.1/30;
            }
        }
    }
}
forwarding-options {
    sampling {
        instance {
            s0 {
                input {
                    rate 1;
                    run-length 0;
                }
                family inet {
                    output {
                        flow-server 198.51.100.2 { # The address of the external
server.
                            port 2055;
                            version9 {
                                template {
                                    v4
```

```
                                }
                            }
                        }
                        interface sp-2/0/0 {
                            source-address 192.168.1.1; # Source address of the
sampled packets
                        }
                    }
                }
            }
            s1 {
                input {
                    rate 1;
                    run-length 0;
                }
                family inet {
                    output {
                        flow-server 198.51.100.2 { # The address of the external
server.
                            port 2055;
                            version9 {
                                template {
                                    v4
                                }
                            }
                        }
                        interface sp-2/1/0 {
                            source-address 192.168.1.2; # Source address of the
sampled packets
                        }
                    }
                }
            }
        }
    }
}

routing-options {
    static {
        route 203.0.113.0/8 next-hop 192.0.2.2;
    }
}
services {
    flow-monitoring {
```

```
        version9 {
            template v4 {
                flow-active-timeout 30;
                flow-inactive-timeout 30;
                ipv4-template;
            }
        }
    }
}
```

Configuration Commands Used for the Configuration Example

The following **set** commands are used for the configuration of the sampling instance in this example. Replace the values in these commands with values relevant to your own network.

- **set chassis fpc 0 sampling-instance s0**
- **set chassis fpc 3 sampling-instance s1**
- **set interfaces ge-0/1/0 unit 0 family inet sampling input**
- **set interfaces ge-0/1/0 unit 0 family inet address**
- **set interfaces ge-1/0/0 unit 0 family inet address**
- **set interfaces sp-2/0/0 unit 0 family inet**
- **set interfaces sp-2/1/0 unit 0 family inet**
- **set interfaces ge-3/1/0 unit 0 family inet sampling input**
- **set interfaces ge-3/1/0 unit 0 family inet address**
- **set forwarding-options sampling instance s0 input rate 1**
- **set forwarding-options sampling instance s0 input run-length 0**
- **set forwarding-options sampling instance s0 family inet output flow-server 198.51.100.2 port 2055**
- **set forwarding-options sampling instance s0 family inet output flow-server 198.51.100.2 version9 template v4;**
- **set forwarding-options sampling instance s0 family inet output interface sp-2/0/0 source-address 192.168.1.1**
- **set forwarding-options sampling instance s1 input rate 1**
- **set forwarding-options sampling instance s1 input run-length 0**
- **set forwarding-options sampling instance s1 family inet output flow-server 198.51.100.2 port 2055**

- **set forwarding-options sampling instance s1 family inet output flow-server 198.51.100.2 version9 template v4;**
- **set forwarding-options sampling instance s1 family inet output interface sp-2/1/0 source-address 192.168.1.2**
- **set routing-options static route 203.0.113.0/8 next-hop 192.0.2.2**
- **set services flow-monitoring version9 template v4 flow-active-timeout 30**
- **set services flow-monitoring version9 template v4 flow-inactive-timeout 30**
- **set services flow-monitoring version9 template v4 ipv4-template**

Verifying Your Work

To verify that your configuration is working as expected, use the following commands on the router that is configured with the sampling instance:

- **show services accounting aggregation template template-name *template-name***
- **show services accounting flow**

The following shows the output of the **show** commands issued on the MX480 router used in this configuration example:

```
user@MX480-router> show services accounting aggregation template template-name v4
                                Src   Dst
                               Port/ Port/
Source          Destination     ICMP  ICMP                          Packet
Address         Address         Type  Code  Proto TOS                Count
10.0.0.6        203.0.113.3      100  1000     17   8                   14
10.0.0.5        203.0.113.2      100  1000     17   8                   15
10.0.0.3        203.0.113.3      100  1000     17   8                   15
10.0.0.2        203.0.113.3      100  1000     17   8                   15
10.0.0.4        203.0.113.2      100  1000     17   8                   15
10.0.0.6        203.0.113.2      100  1000     17   8                   15
10.0.0.4        203.0.113.3      100  1000     17   8                   15
10.0.0.2        203.0.113.2      100  1000     17   8                   16
10.0.0.3        203.0.113.2      100  1000     17   8                   15
10.0.0.5        203.0.113.3      100  1000     17   8                   15

user@MX480-router> show services accounting aggregation template template-name v4
                                Src   Dst
                               Port/ Port/
Source          Destination     ICMP  ICMP                          Packet
Address         Address         Type  Code  Proto TOS                Count
```

```
10.0.0.6        203.0.113.3      100  1000     17   8                  16
10.0.0.5        203.0.113.2      100  1000     17   8                  17
10.0.0.3        203.0.113.3      100  1000     17   8                  16
10.0.0.2        203.0.113.3      100  1000     17   8                  16
10.0.0.4        203.0.113.2      100  1000     17   8                  17
10.0.0.6        203.0.113.2      100  1000     17   8                  17
10.0.0.4        203.0.113.3      100  1000     17   8                  16
10.0.0.2        203.0.113.2      100  1000     17   8                  17
10.0.0.3        203.0.113.2      100  1000     17   8                  17
10.0.0.5        203.0.113.3      100  1000     17   8                  16

user@MX480-router> show services accounting flow
  Flow information
    Interface name: sp-2/0/0, Local interface index: 152
    Flow packets: 884, Flow bytes: 56576
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 628
    Active flows: 10, Total flows: 35
    Flows exported: 75, Flows packets exported: 14
    Flows inactive timed out: 25, Flows active timed out: 75

user@MX480-router> show services accounting flow
  Flow information
    Interface name: sp-2/0/0, Local interface index: 152
    Flow packets: 898, Flow bytes: 57472
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 628
    Active flows: 10, Total flows: 35
    Flows exported: 75, Flows packets exported: 14
    Flows inactive timed out: 25, Flows active timed out: 75
```

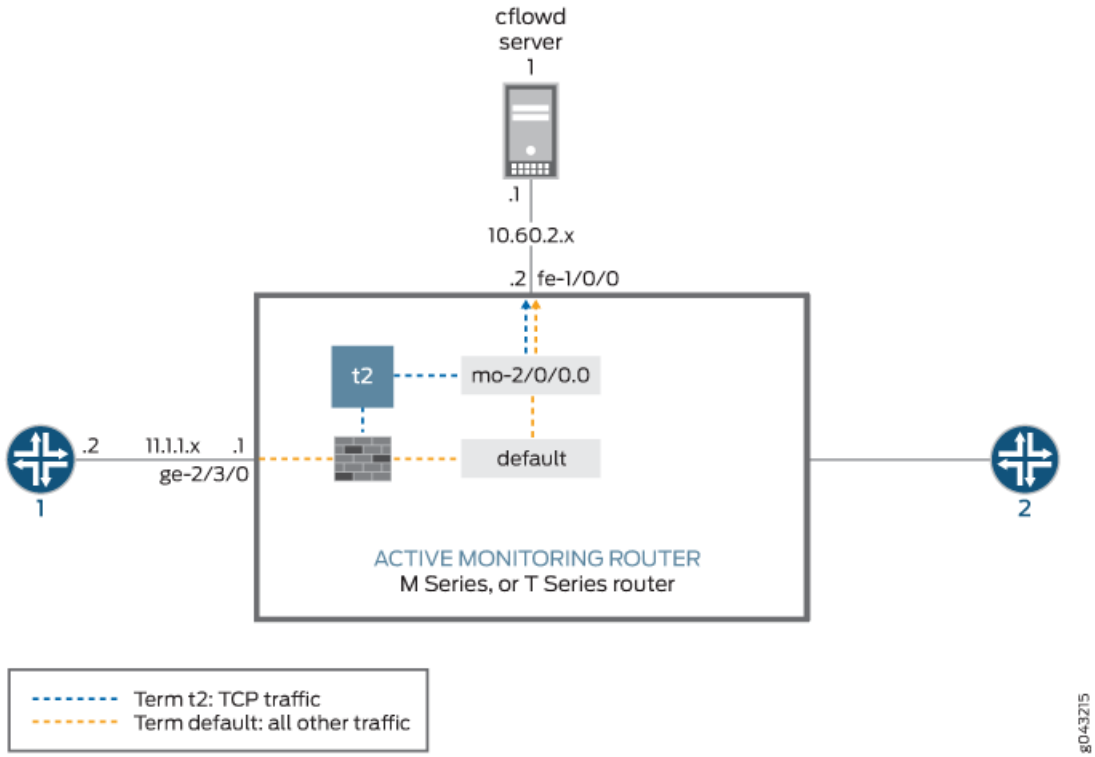
RELATED DOCUMENTATION

- Configuring Sampling Instance on MX, M and T Series Routers or QFX Series Switches | 388
- Configuring Active Flow Monitoring | 34
- sampling-instance | 1130

Example: Sampling and Discard Accounting Configuration on M, MX and T Series Routers

Discard accounting allows you to sample traffic, send it to a flow server for analysis, and discard all packets without forwarding them to their intended destination. Discard accounting is enabled with the **discard accounting *group-name*** statement in a firewall filter at the **[edit firewall family inet filter *filter-name* term *term-name* then]** hierarchy level. Then, the filter is applied to an interface with the **filter** statement at the **[edit interfaces *interface-name* unit *unit-number* family inet]** hierarchy level and processed with the **output** statement at the **[edit forwarding-options accounting *group-name*]** hierarchy level.

Figure 24: Active Flow Monitoring—Sampling and Discard Accounting Topology Diagram



In Figure 19 on page 126, traffic from Router 1 arrives on the monitoring router's Gigabit Ethernet **ge-2/3/0** interface. The export interface leading to the flow server is **fe-1/0/0** and there is no exit interface.

In this example, TCP traffic is sent to one accounting group and all other traffic is diverted to a second group. After being sampled and counted, the two types of traffic are acted upon by the sampling and accounting processes. These processes create flow records and send the records to the version 8 flow server for analysis. Because multiple types of traffic are sent to the same server, we recommend that you configure the **engine-id**, **engine-type**, and **source-address** statements manually in your accounting and

sampling hierarchies. This way, you can differentiate between traffic types when they arrive at the flow server.

```
[edit]
interfaces {
    sp-2/0/0 { # This adaptive services interface creates the flow records.
        unit 0 {
            family inet {
                address 10.5.5.1/32 {
                    destination 10.5.5.2;
                }
            }
        }
    }
    fe-1/0/0 { # This is the interface where records are sent to the flow server.
        unit 0 {
            family inet {
                address 10.60.2.2/30;
            }
        }
    }
    ge-2/3/0 { # This is the input interface where traffic enters the router.
        unit 0 {
            family inet {
                filter {
                    input catch_all;
                }
                address 10.1.1.1/20;
            }
        }
    }
}
forwarding-options {
    sampling { # The router samples the traffic.
        input {
            rate 100; # One out of every 100 packets is sampled.
            }
        }
            family inet {
                output { # The sampling process creates and exports flow records.
                flow-server 10.60.2.1 { # You can configure a variety of settings.
                    port 2055;
                    version 8;
                    aggregation { # Aggregation is unique to flow version 8.
                        protocol-port;
```

```
            source-destination-prefix;
          }
        }
        aggregate-export-interval 90;
        flow-inactive-timeout 60;
        flow-active-timeout 60;
        interface sp-2/0/0 { # This statement enables PIC-based sampling.
          engine-id 5; # Engine statements are dynamic, but can be configured.
          engine-type 55;
          source-address 10.60.2.2; # You must configure this statement.
        }
      }
}
accounting counter1 { # This discard accounting process handles default traffic.
  output { # This process creates and exports flow records.
    flow-inactive-timeout 65;
    flow-active-timeout 65;
    flow-server 10.60.2.1 { # You can configure a variety of settings.
      port 2055;
      version 8;
      aggregation { # Aggregation is unique to version 8.
        protocol-port;
        source-destination-prefix;
      }
    }
    interface sp-2/0/0 {  # This statement enables PIC-based discard accounting.
      engine-id 1; # Engine statements are dynamic, but can be configured.
      engine-type 11;
      source-address 10.60.2.3; # You must configure this statement.
    }
  }
}
accounting t2 { # The second discard accounting process handles the TCP traffic.
  output { # This process creates and exports flow records.
    aggregate-export-interval 90;
    flow-inactive-timeout 65;
    flow-active-timeout 65;
    flow-server 10.60.2.1 { # You can configure a variety of settings for the server.
      port 2055;
      version 8;
      aggregation { # Aggregation is unique to version 8.
        protocol-port;
        source-destination-prefix;
      }
```

```
        }
        interface sp-2/0/0 { # This statement enables PIC-based discard accounting.
          engine-id 2; # Engine statements are dynamic, but can be configured.
          engine-type 22;
          source-address 10.60.2.4;# You must configure this statement.
        }
      }
    }
}
firewall {
  family inet {
    filter catch_all { # Apply the firewall filter on the input interface.
      term t2 { # This places TCP traffic into one group for sampling and
        from { # discard accounting.
          protocol tcp;
        }
        then {
          count c2;# The count action counts traffic as it enters the router.
          sample; # The sample action sends the traffic to the sampling process.
          discard accounting t2; # The discard accounting discards traffic.
        }
      }
      term default { # Performs sampling and discard accounting on all other traffic.
        then {
          count counter; # The count action counts traffic as it enters the router.
          sample# The sample action sends the traffic to the sampling process.
          discard accounting counter1; # This activates discard accounting.
        }
      }
    }
  }
}
```

Verifying Your Work

To verify that your configuration is correct, use the following commands on the monitoring station that is configured for active flow monitoring:

- **show services accounting aggregation** (for version 8 flows only)
- **show services accounting errors**
- **show services accounting (flow | flow-detail)**
- **show services accounting memory**

- **show services accounting packet-size-distribution**
- **show services accounting status**
- **show services accounting usage**

The following shows the output of the **show** commands used with the configuration example:

user@host> **show services accounting flow name t2**

```
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: t2
  Flow information
    Flow packets: 56130820, Flow bytes: 3592372480
    Flow packets 10-second rate: 13024, Flow bytes 10-second rate: 833573
    Active flows: 600, Total flows: 600
    Flows exported: 28848, Flows packets exported: 960
    Flows inactive timed out: 0, Flows active timed out: 35400

user@host> show services accounting
Service Name:
    (default sampling)
    counter1
    t2

user@host> show services accounting aggregation protocol-port detail name t2
Service Accounting interface: sp-2/0/0, Local interface index: 468
Service name: t2

  Protocol: 6, Source port: 20, Destination port: 20
  Start time: 442794, End time: 6436260
  Flow count: 1, Packet count: 4294693925, Byte count: 4277471552

user@host> show services accounting aggregation source-destination-prefix name
t2 limit 10 order packets
Service Accounting interface: sp-2/0/0, Local interface index: 542
Service name: t2
     Source     Destination  Input SNMP  Output SNMP   Flow  Packet    Byte
     Prefix          Prefix       Index        Index  count   count   count
10.1.1.2/20    10.225.0.1/0          24           26      0      13    9650
10.1.1.2/20   10.143.80.1/0          24           26      0      13   10061
10.1.1.2/20   10.59.176.1/0          24           26      0      13   10426
10.1.1.2/20     10.5.32.1/0          24           26      0      13   12225
10.1.1.2/20    10.36.16.1/0          24           26      0      13    9116
10.1.1.2/20     10.1.96.1/0          24           26      0      12   11050
10.1.1.2/20    10.14.48.1/0          24           26      0      13   10812
```

```
10.1.1.2/20   10.31.192.1/0          24           26      0      13   11473
10.1.1.2/20  10.129.144.1/0          24           26      0      13    7647
10.1.1.2/20  10.188.160.1/0          24           26      0      13   10056

user@host> show services accounting aggregation source-destination-prefix name
t2 extensive limit 3
Service Accounting interface: sp-2/0/0, Local interface index: 542
Service name: t2

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 10.200.176.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 5340

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 10.243.160.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 5490

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 10.162.160.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 4079
```

CHAPTER 3

Monitoring Traffic Using Passive Flow Monitoring

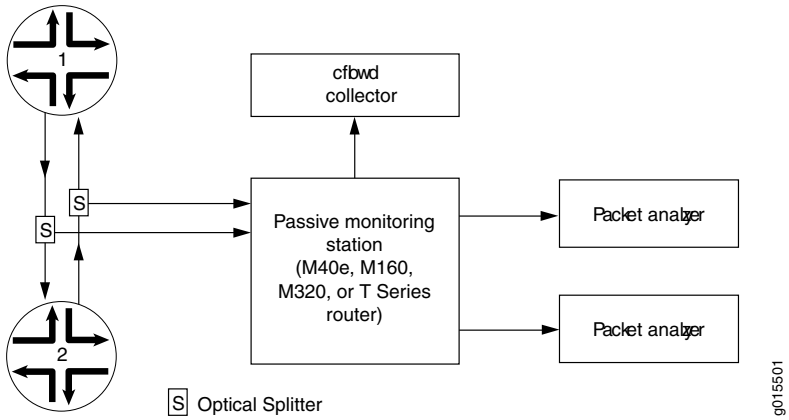
IN THIS CHAPTER

Understanding Passive Flow Monitoring on T Series, M Series and MX Series Routers

Flow monitoring version 5 supports passive flow monitoring. Versions 8 and 9 do not support passive flow monitoring.

The M40e, M160, M320, MX Series, or T Series router that is used for passive flow monitoring does not route packets from monitored interfaces, nor does it run any routing protocols related to those interfaces; it only passes along intercepted traffic and receives traffic flows. Figure 25 on page 197 shows a typical topology for the passive flow monitoring application.

Figure 25: Passive Flow Monitoring Application Topology



Traffic travels normally between Router 1 and Router 2. To redirect IPv4 traffic, you insert an optical splitter on the interface between these two routers. The optical splitter copies and redirects the traffic to the monitoring station. The optical cable connects only the receive port on the monitoring station, never the transmit port. This configuration allows the monitoring station to receive traffic only from the router being monitored but never to transmit it back.

If you are monitoring traffic flow, the Internet Processor II ASIC in the router forwards a copy of the traffic to the Monitoring Services or Monitoring Services II PIC in the monitoring station. If there is more than one Monitoring Services PIC installed, the monitoring station distributes the load of the incoming traffic across the multiple PICs. The Monitoring Services PICs generate flow records in version 5 format, and the records are exported to the flow collector.

When you are performing lawful interception of packets, the Internet Processor II ASIC filters the incoming traffic and forwards it to the Tunnel Services PIC. Filter-based forwarding is then applied to direct the traffic to the packet analyzers. Optionally, the intercepted traffic or the flow records can be encrypted by the ES PIC and then sent to their destination. With additional configuration, flow records can be processed by a flow collector and flows can be captured dynamically.

With MPLS passive monitoring, the router can process MPLS packets with label values that do not have corresponding entries in the **mpls.0** routing table. You can divert these unrecognized MPLS packets, remove the MPLS labels, and redirect the underlying IPv4 packets. This is equivalent to a default route for MPLS packets or a promiscuous label. Because this application does not use a Monitoring Services PIC, see the *Junos MPLS Applications Configuration Guide* for more information about MPLS passive monitoring.

RELATED DOCUMENTATION

Active Flow Monitoring Overview | 45
Active Flow Monitoring Overview | 44

Passive Flow Monitoring Overview

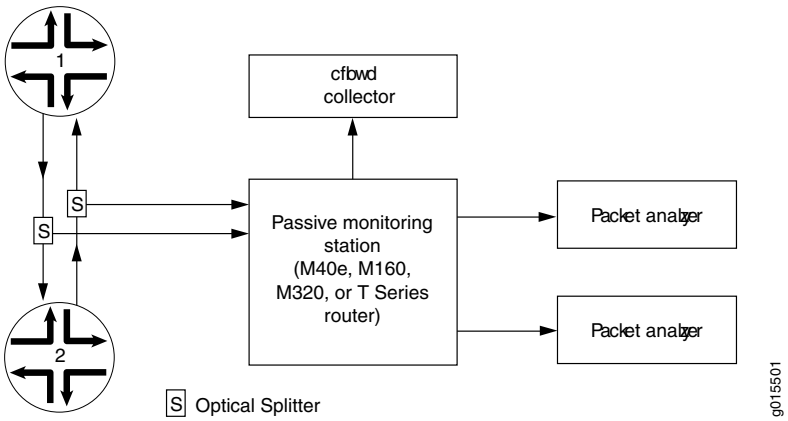
Using a Juniper Networks M Series Multiservice Edge or T Series Core router, a selection of PICs (including the Monitoring Services PIC, Adaptive Services [AS] PIC, Multiservices PIC, or Multiservices DPC) and other networking hardware, you can monitor traffic flow and export the monitored traffic. Monitoring traffic allows you to do the following:

- Gather and export detailed information about IP version 4 (IPv4) traffic flows between source and destination nodes in your network.
- Sample all incoming IPv4 traffic on the monitoring interface and present the data in cflowd record format.
- Perform discard accounting on an incoming traffic flow.
- Encrypt or tunnel outgoing cflowd records, intercepted IPv4 traffic, or both.
- Direct filtered traffic to different packet analyzers and present the data in its original format (port mirror).

> **NOTE:** Monitoring Services PICs, AS PICs, and Multiservices PICs must be mounted on an Enhanced Flexible PIC Concentrator (FPC) in an M Series or T Series router.
>
> Multiservices DPCs installed in Juniper Networks MX Series routers support the same functionality, with the exception of the passive monitoring and flow-tap features.

The router used for passive monitoring does not route packets from the monitored interface, nor does it run any routing protocols related to those interfaces; it only receives traffic flows, collects intercepted traffic, and exports it to cflowd servers and packet analyzers. Figure 26 on page 199 shows a typical topology for the passive flow-monitoring application.

Figure 26: Passive Monitoring Application Topology



Traffic travels normally between Router 1 and Router 2. To redirect IPv4 traffic, you insert an optical splitter on the interface between these two routers. The optical splitter copies and redirects the traffic to the monitoring station, which is an M40e, M160, M320, or T Series router. The optical cable connects only the receive port on the monitoring station, never the transmit port. This configuration allows the monitoring station to receive traffic from the router being monitored but never to transmit it back.

If you are monitoring traffic flow, the Internet Processor II application-specific integrated circuit (ASIC) in the router forwards a copy of the traffic to the Monitoring Services, Adaptive Services, or Multiservices PIC in the monitoring station. If more than one monitoring PIC is installed, the monitoring station distributes the load of the incoming traffic across the multiple PICs. The monitoring PICs generate flow records in cflowd version 5 format, and the records are then exported to the cflowd collector.

If you are performing lawful interception of traffic between the two routers, the Internet Processor II ASIC filters the incoming traffic and forwards it to the Tunnel Services PIC. Filter-based forwarding is then applied to direct the traffic to the packet analyzers.

Optionally, the intercepted traffic or the cflowd records can be encrypted by the ES PIC or IP Security (IPsec) services and then sent to a cflowd server or packet analyzer.

RELATED DOCUMENTATION

Enabling Passive Flow Monitoring on M Series, MX Series or T Series Routers | 203

Passive Flow Monitoring System Requirements for T Series, M Series and MX Series Routers

To perform passive flow monitoring, your router must meet these minimum requirements:

- Junos OS Release 9.2 or later for passive flow monitoring support for IQ2 interfaces only on M120, M320, T320, T640, T1600 and MX-series routers.
- Junos OS Release 8.5 or later for passive flow monitoring support on the MX Series MultiServices routers
- Junos OS Release 8.4 or later for passive flow monitoring support on the MultiServices 400 PIC (Type 2)
- Junos OS Release 7.6 or later to clear error and flow statistics with the **clear passive-monitoring statistics** command
- Junos OS Release 7.5 or later for support of the dynamic flow capture (DFC) Management Information Base (MIB)
- Junos OS Release 7.4 or later for dynamic flow capture on Monitoring Services III PICs installed in T Series and M320 routers, and port mirroring of IPv6 packets
- Junos OS Release 7.3 or later for passive flow monitoring on selected Ethernet-based interfaces and filter-based forwarding on output interfaces
- Junos OS Release 7.1 or later for passive flow monitoring and flow collection services on Monitoring Services II PICs installed in T Series and M320 routers
- Junos OS Release 6.4 or later for support of the next-hop IP address field in flow monitoring version 5 records
- Junos OS Release 6.2 or later for ATM2 intelligent queuing (IQ) interface passive monitoring, flow collection services, and MPLS label stripping
- Junos OS Release 6.1 or later for MPLS passive monitoring
- Junos OS Release 6.0 or later for the Monitoring Services II PIC
- Junos OS Release 5.7 or later for the automatic insertion of autonomous system (AS) numbers and SNMP index values for interfaces into flow records
- Junos OS Release 5.4 or later for the Monitoring Services PIC
- M40e, M160, M320, MX Series, or T Series router with an Internet Processor II ASIC or later
- Type 1 enhanced FPCs
- Two optical splitters
- A Tunnel Services PIC (required if you wish to send traffic to more than one analyzer)
- An input interface from the following list:

  - SONET/SDH PIC—OC3, OC12, or OC48
  - ATM2 IQ PIC—OC3 or OC12
  - 4-port Fast Ethernet PIC
  - Gigabit Ethernet PIC—4-port with small form-factor pluggable transceiver (SFP) or 10-port with SFP
  - 1-port 10-Gigabit Ethernet PIC with XENPAK
- Outgoing PICs to connect to the flow collector or packet analyzer
- Flow monitoring version 5 collector
- ES PIC and packet analyzers (optional)

RELATED DOCUMENTATION

Active Flow Monitoring System Requirements | 38

Active Flow Monitoring PIC Specifications | 40

Passive Flow Monitoring Router and Software Considerations for T Series, M Series and MX Series Routers

There are several hardware and software considerations when you implement passive flow monitoring. When defining the hardware requirements of the monitoring station, keep in mind the following:

- The input interfaces on the monitoring station must be SONET/SDH interfaces (OC3, OC12, or OC48), ATM2 IQ interfaces (OC3 or OC12), 4-port Fast Ethernet interfaces, Gigabit Ethernet interfaces with SFP (4-port or 10-port), or 1-port 10-Gigabit Ethernet interfaces with XENPAK.
- To monitor the flows in both directions for a single interface, the monitoring station must have two SONET/SDH, ATM2 IQ, or Ethernet-based receive ports, one for each direction of flow. In Figure 25 on page 197, the monitoring station needs one port to monitor the traffic flowing from Router 1 to Router 2, and a second port to monitor the traffic flowing from Router 2 to Router 1.
- The Monitoring Services PICs must be installed in a Type 1 enhanced FPC slot.
- Type 1 and Type 2 Tunnel Services PICs are supported.
- Use an ES PIC to encrypt the flow export.

When defining a traffic monitoring strategy, keep in mind the following:

- The monitoring station collects only IPv4 packets. All other packet formats are discarded and not counted.

- You can set the amount of time a data flow can be inactive before the monitoring station terminates the flow and exports the flow data. To set the timer, include the **flow-inactive-timeout** statement at the **[edit forwarding-options monitoring *group-name* family inet output]** hierarchy level. The timer value can be from 15 seconds through 1800 seconds, with a default value of 60 seconds.

You can also configure the monitoring station to collect periodic flow reports for flows that last longer than the configured active timeout. To set this activity timer, include the **flow-active-timeout** statement at the **[edit forwarding-options monitoring *group-name* family inet output]** hierarchy level. The timer value can be from 60 seconds through 1800 seconds, with a default value of 180 seconds.

- Multiple expired flows are exported together, if possible. A UDP packet is sent when one of the following conditions is met:
  - When 30 flows are contained in the current packet, the flows are exported.
  - If there are fewer than 30 flows but the export timer expires, the flows are exported one second after the timer expires.
- TCP and UDP flows are considered differently:
  - TCP flows watch for a segment containing the **FIN** bit and a subsequent acknowledgement (**ACK**) to detect the end of a flow. Alternately, a TCP reset (**RST**) can also indicate the end of a flow. When these TCP combinations are detected, the flow expires. The **FIN+ACK** and **RST** cases cover most TCP stream closures. For all other flows, an inactive timeout is needed.
  - All non-TCP flows, such as UDP, depend on timeout mechanisms for export.
- The default MTU value for SONET/SDH interfaces is 4474 bytes; for Gigabit Ethernet and Fast Ethernet interfaces, it is 1500 bytes. If the monitoring station receives packets exceeding 4474 bytes, they are discarded; no fragmentation is performed. Note that the supported MTU size on the Gigabit Ethernet or Fast Ethernet PICs might exceed 1500 bytes, depending on the type of PIC.
- Any incoming traffic that is discarded is not forwarded to packet analyzers.
- The interfaces on the monitoring station that collect intercepted traffic must be configured with Cisco HDLC or PPP encapsulation.
- You must always use a standard interface (for example, one that follows the usual ***interface-name-fpc/pic/slot*** format) to send flow records to a flow server. Flow data generated by the Monitoring Services or Monitoring Services II PICs will not be delivered to the server across the **fxp0** interface.
- You can send version 5 records to multiple flow servers. You can configure up to eight servers and flow traffic is load-balanced between the servers in a round-robin fashion. If one of the servers ceases operation, flow traffic load-balances automatically between the remaining active servers. To configure, include up to eight **flow-server** statements at the **[edit forwarding-options monitoring *group-name* output]** hierarchy level.

Enabling Passive Flow Monitoring on M Series, MX Series or T Series Routers

IN THIS SECTION

You can monitor IPv4 traffic from another router if you have the following components installed in an M Series, MX Series, or T Series router:

- Monitoring Services, Adaptive Services, or Multiservices PICs to perform the service processing
- SONET/SDH, Fast Ethernet, or Gigabit Ethernet PICs as transit interface

On SONET/SDH interfaces, you enable passive flow monitoring by including the **passive-monitor-mode** statement at the [edit interfaces so-*fpc/pic/port* unit *logical-unit-number*] hierarchy level:

```
[edit interfaces so-fpc/pic/port unit logical-unit-number]
passive-monitor-mode;
```

On Asynchronous Transfer Mode (ATM), Fast Ethernet, or Gigabit Ethernet interfaces, you enable passive flow monitoring by including the **passive-monitor-mode** statement at the **[edit interfaces *interface-name*]** hierarchy level:

```
[edit interfaces interface-name]
passive-monitor-mode;
```

IPv6 passive monitoring is not supported on Monitoring Services PICs. You must configure port mirroring to forward the packets from the passive monitored ports to other interfaces. Interfaces configured on the following FPCs and PIC support IPv6 passive monitoring on the T640 and T1600 Series routers:

- Enhanced Scaling FPC2
- Enhanced Scaling FPC3
- Enhanced II FPC1
- Enhanced II FPC2
- Enhanced II FPC3
- Enhanced Scaling FPC4

- Enhanced Scaling FPC4.1
- 4-port 10-Gigabit Ethernet LAN/WAN PIC with XFP (supported on both WAN-PHY and LAN-PHY mode for both IPv4 and IPv6 addresses)
- Gigabit Ethernet PIC with SFP
- 10-Gigabit Ethernet PIC with XENPAK (T1600 Series router)
- SONET/SDH OC192/STM64 PIC (T1600 Series router)
- SONET/SDH OC192/STM64 PICs with XFP (T1600 Series router)
- SONET/SDH OC48c/STM16 PIC with SFP (T1600 Series router)
- SONET/SDH OC48/STM16 (Multi-Rate)
- SONET/SDH OC12/STM4 (Multi–Rate) PIC with SFP
- Type 1 SONET/SDH OC3/STM1 (Multi–Rate) PIC with SFP

To configure port mirroring, include the **port-mirroring** statement at the **[edit forwarding-options]** hierarchy level.

When you configure an interface in passive monitoring mode, the Packet Forwarding Engine silently drops packets coming from that interface and destined to the router itself. Passive monitoring mode also stops the Routing Engine from transmitting any packet from that interface. Packets received from the monitored interface can be forwarded to monitoring interfaces. If you include the **passive-monitor-mode** statement in the configuration:

- The ATM interface is always up, and the interface does not receive or transmit incoming control packets, such as Operation, Administration, and Maintenance (OAM) and Interim Local Management Interface (ILMI) cells.
- The SONET/SDH interface does not send keepalives or alarms and does not participate actively on the network.
- Gigabit and Fast Ethernet interfaces can support both per-port passive monitoring and per-VLAN passive monitoring. The destination MAC filter on the receive port of the Ethernet interfaces is disabled.
- Ethernet encapsulation options are not allowed.
- Ethernet interfaces do not support the **stacked-vlan-tagging** statement for both IPv4 and IPv6 packets in passive monitoring mode.

On monitoring services interfaces, you enable passive flow monitoring by including the **family** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level, specifying the **inet** option:

```
[edit interfaces interface-name unit logical-unit-number]
family inet;
```

For the monitoring services interface, you can configure multiservice physical interface properties. For more information, see "Configuring Flow-Monitoring Interfaces" on page 4.

For conformity with the cflowd record structure, you must include the **receive-options-packets** and **receive-ttl-exceeded** statements at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet]** hierarchy level:

```
[edit interfaces interface-name unit logical-unit-number family inet]
receive-options-packets;
receive-ttl-exceeded;
```

For more information, see the following sections:

Passive Flow Monitoring for MPLS Encapsulated Packets

On monitoring services interfaces, you can process MPLS packets that have not been assigned label values and have no corresponding entry in the **mpls.0** routing table. This allows you to assign a default route to unlabeled MPLS packets.

To configure a default label value for MPLS packets, include the **default-route** statement at the **[edit protocols mpls interface *interface-name* label-map]** hierarchy level:

```
[edit protocols mpls interface interface-name label-map]
default-route {
    (next-hop (address | interface-name | address/interface-name)) | (reject | discard);
    (pop | (swap <out-label>);
    class-of-service value;
    preference preference;
    type type;
}
```

For more information about static labels, see the *MPLS Applications User Guide*.

Removing MPLS Labels from Incoming Packets

The Junos OS can forward only IPv4 packets to a Monitoring Services, Adaptive Services, or Multiservices PIC. IPv4 and IPv6 packets with MPLS labels cannot be forwarded to a monitoring PIC. By default, if packets with MPLS labels are forwarded to the monitoring PIC, they are discarded. To monitor IPv4 and IPv6 packets with MPLS labels, you must remove the MPLS labels as the packets arrive on the interface.

You can remove MPLS labels from an incoming packet by including the **pop-all-labels** statement at the **[edit interfaces *interface-name* (atm-options | fastether-options | gigether-options | sonet-options) mpls]** hierarchy level:

```
[edit interfaces interface-name (atm-options | fastether-options | gigether-options | sonet-options) mpls]
pop-all-labels {
    required-depth [ numbers ];
}
```

For MX Series routers with MPCs, the **pop-all-labels** statement pops all labels by default and the **required-depth** statement is ignored.

For other configurations, you can remove up to two MPLS labels from an incoming packet. By default, the **pop-all-labels** statement takes effect for incoming packets with one or two labels. You can specify the number of MPLS labels that an incoming packet must have for the **pop-all-labels** statement to take effect by including the **required-depth** statement at the **[edit interfaces *interface-name* (atm-options | fastether-options | gigether-options | sonet-options) mpls pop-all-labels]** hierarchy level:

```
[edit interfaces interface-name (atm-options | fastether-options | gigether-options | sonet-options) mpls
    pop-all-labels]
required-depth [ numbers ];
```

The required depth can be **1**, **2**, or **[1 2]**. If you include the **required-depth 1** statement, the **pop-all-labels** statement takes effect for incoming packets with one label only. If you include the **required-depth 2** statement, the **pop-all-labels** statement takes effect for incoming packets with two labels only. If you include the **required-depth [1 2]** statement, the **pop-all-labels** statement takes effect for incoming packets with one or two labels. A required depth of **[1 2]** is equivalent to the default behavior of the **pop-all-labels** statement.

When you remove MPLS labels from incoming packets, note the following:

- The **pop-all-labels** statement has no effect on IP packets with three or more MPLS labels except for MX Series routers with MPCs.
- When you enable MPLS label removal, you must configure all ports on a PIC with the same label popping mode and required depth.
- You use the **pop-all-labels** statement to enable passive monitoring applications, not active monitoring applications.
- You cannot apply MPLS filters or accounting to the MPLS labels because the labels are removed as soon as the packet arrives on the interface.
- On ATM2 interfaces, you must use a label value greater than 4095 because the lower range of MPLS labels is reserved for label-switched interface (LSI) and virtual private LAN service (VPLS) support. For more information, see the *Junos OS VPNs Library for Routing Devices*.
- The following ATM encapsulation types are not supported on interfaces with MPLS label removal:
  - **atm-ccc-cell-relay**
  - **atm-ccc-vc-mux**

- **atm-mlppp-llc**
- **atm-tcc-snap**
- **atm-tcc-vc-mux**
- **ether-over-atm-llc**
- **ether-vpls-over-atm-llc**

Example: Enabling IPv4 Passive Flow Monitoring

The following example shows a complete configuration for enabling passive flow monitoring on an Ethernet interface.

In this example, the Gigabit Ethernet interface can accept all Ethernet packets. It strips VLAN tags (if there are any) and up to two MPLS labels blindly, and passes IPv4 packets to the monitoring interface. With this configuration, it can monitor IPv4, VLAN+IPv4, VLAN+MPLS+IPv4, and VLAN+MPLS+MPLS+IPv4 labeled packets.

The Fast Ethernet interface can accept only packets with VLAN ID 100. All other packets are dropped. With this configuration, it can monitor VLAN (ID=100)+IPv4, VLAN (ID=100)+MPLS+IPv4, and VLAN (ID=100)+MPLS+MPLS+IPv4 labeled packets.

```
[edit firewall]
family inet {
    filter input-monitoring-filter {
        term def {
            then {
                count counter;
                accept;
            }
        }
    }
}
[edit interfaces]
ge-0/0/0 {
    passive-monitor-mode;
    gigether-options {
        mpls {
            pop-all-labels;
        }
    }
    unit 0 {
        family inet {
            filter {
```

```
            input input-monitoring-filter;
          }
        }
      }
    }
    fe-0/1/0 {
      passive-monitor-mode;
      vlan-tagging;
      fastether-options {
        mpls {
          pop-all-labels required-depth [ 1 2 ];
        }
      }
      unit 0 {
        vlan-id 100;
        family inet {
          filter {
            input input-monitoring-filter;
          }
        }
      }
    }
    mo-1/0/0 {
      unit 0 {
        family inet {
          receive-options-packets;
          receive-ttl-exceeded;
        }
      }
      unit 1 {
        family inet;
      }
    }
    [edit forwarding-options]
    monitoring mon1 {
      family inet {
        output {
          export-format cflowd-version-5;
          cflowd 192.0.2.2 port 2055;
          interface mo-1/0/0.0 {
            source-address 192.0.2.1;
          }
        }
      }
```

```
}
[edit routing-instances]
monitoring-vrf {
    instance-type vrf;
    interface ge-0/0/0.0;
    interface fe-0/1/0.0;
    interface mo-1/0/0.1;
    route-distinguisher 68:1;
    vrf-import monitoring-vrf-import;
    vrf-export monitoring-vrf-export;
    routing-options {
        static {
            route 0.0.0.0/0 next-hop mo-1/0/0.1;
        }
    }
}
[edit policy-options]
policy-statement monitoring-vrf-import {
    then {
        reject;
    }
}
policy-statement monitoring-vrf-export {
    then {
        reject;
    }
}
```

Example: Enabling IPv6 Passive Flow Monitoring

The following example shows a complete configuration for enabling IPv6 passive flow monitoring on an Ethernet interface.

In this example, the Gigabit Ethernet interface can accept all Ethernet packets. It strips VLAN tags (if there are any) and up to two MPLS labels blindly, and passes IPv6 packets to the monitoring interface. With this configuration, the Gigabit Ethernet interface can monitor IPv6, VLAN+IPv6, VLAN+MPLS+IPv6, and VLAN+MPLS+MPLS+IPv6 labeled packets.

The vlan-tagged Gigabit Ethernet interface can accept only packets with VLAN ID 100. All other packets are dropped. With this configuration, it can monitor VLAN (ID=100)+IPv6, VLAN (ID=100)+MPLS+IPv6, and VLAN (ID=100)+MPLS+MPLS+IPv6 labeled packets.

```
[edit interfaces]
```

```
xe-0/1/0 {
    passive-monitor-mode;
    unit 0 {
        family inet6 {
            filter {
                input port-mirror6;
            }
            address 2001:db8::1/128;
        }
    }
}
    xe-0/1/2 {
        passive-monitor-mode;
        vlan-tagging;
        unit 0 {
            vlan-id 100;
            family inet6 {
                filter {
                    input port-mirror6;
                }
            }
        }
    }
    xe-0/1/1 {
        unit 0 {
            family inet6 {
                address 2001:db8::1/128;
            }
        }
    }
[edit firewall]
family inet6 {
    filter port-mirror6 {
        term term2 {
            then {
                count count_pm;
                port-mirror;
                accept;
            }
        }
    }
}
[edit forwarding options]
port-mirroring {
```

```
    input {
        rate 1;
    }
    family inet6 {
        output {
            interface xe-0/1/1.0 {
                next-hop 2001:db8::3;
            }
            no-filter-check;
        }
    }
}
```

RELATED DOCUMENTATION

Passive Flow Monitoring Overview | 198

Configuring Passive Flow Monitoring

Table 28 on page 211 shows which Juniper Networks PICs and routers support passive flow monitoring. The PICs receive passively monitored network traffic from an input interface (SONET/SDH, ATM2 IQ, Fast Ethernet, Gigabit Ethernet, or 10-Gigabit Ethernet), convert the received packets into flow records, and export them to a flow server for further analysis.

Table 28: Passive Flow Monitoring PIC Support

PIC Type	M40e	M160	T Series/ M320
Monitoring Services PIC	Yes	Yes	No
Monitoring Services II PIC	Yes	Yes	Yes
Monitoring Services III PIC	Yes	Yes	Yes
MultiServices 400 PIC (Type 2)	Yes	No	Yes

The key configuration hierarchy statement for passive flow monitoring is the **monitoring** statement found at the **[edit forwarding-options]** hierarchy level. At minimum, you must configure a VRF routing instance to direct the traffic to a monitoring services interface for flow processing.

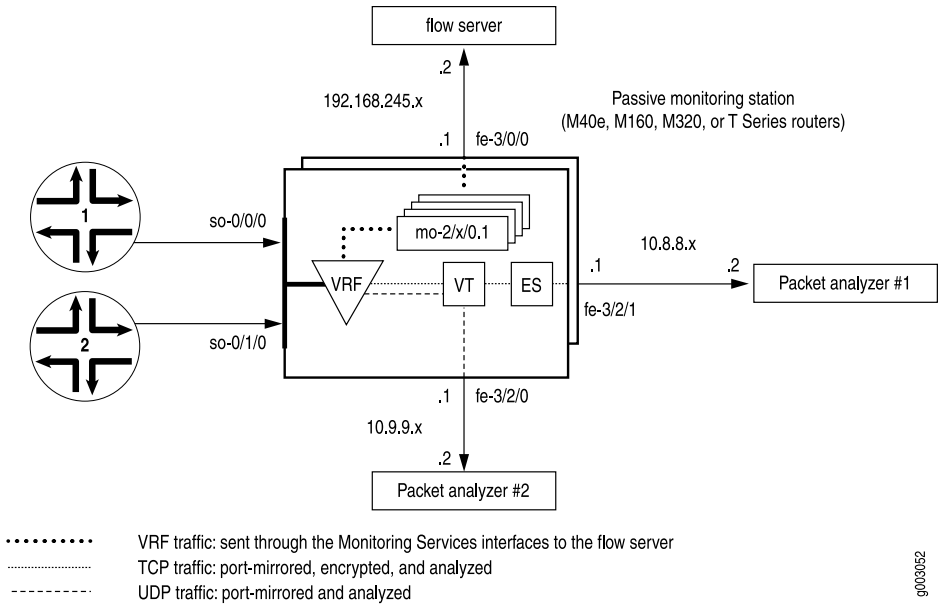
However, there are several options you can use that add complexity to passive flow monitoring. For example, you can configure the router to direct traffic into a routing instance and deliver the traffic into a monitoring group. You can also use port mirroring and filter-based forwarding to copy and redirect traffic. Optionally, you can configure the monitoring station to encrypt flow output before it is sent to a flow server for processing, to send flow records to a flow collector, or to process on-demand monitoring requests with dynamic flow capture.

RELATED DOCUMENTATION

- Copying and Redirecting Traffic with Port Mirroring and Filter-Based Forwarding | **229**
- Using an M, MX or T Series Router Flow Collector Interface to Process and Export Multiple Flow Records | **247**
- Passive Flow Monitoring Router and Software Considerations for T Series, M Series and MX Series Routers | **201**

Example: Passive Flow Monitoring Configuration on M, MX and T Series Routers

Figure 27: Passive Flow Monitoring—Topology Diagram



In Figure 27 on page 212, traffic enters the monitoring station through interfaces **so-0/0/0** and **so-0/1/0**. After the firewall filter accepts the traffic to be monitored, the packets enter a VRF instance.

The original packets travel within the VRF instance to the Monitoring Services PIC for flow processing. The final flow packets are sent from the monitoring services interfaces out the **fe-3/0/0** interface to a flow server.

A copy of the accepted traffic is port-mirrored to the Tunnel PIC. As the copied packets enter the tunnel interface, a second firewall filter separates TCP and UDP packets and places them into two filter-based forwarding instances. The UDP instance directs the UDP packets to a packet analyzer attached to **fe-3/2/0**. The TCP instance sends the TCP packets to the ES PIC for encryption and the ES PIC sends the packets to a second packet analyzer connected to **fe-3/2/1**.

Your first step is to define a firewall filter to select packets for monitoring. All filtered traffic must be accepted, and the **port-mirror** statement at the **[edit firewall family inet filter *filter-name* term *term-name* then]** hierarchy level facilitates port mirroring.

Next, configure the input SONET/SDH interfaces and apply the firewall filter that you just defined. The **passive-monitor-mode** statement disables SONET keepalives on the SONET/SDH interfaces and enables passive flow monitoring.

Configure all other interfaces that you will use with the monitoring application, including the monitoring services interfaces, the export interfaces, the tunnel interface, and the ES interface. Once the interfaces are in place, configure a VRF instance and monitoring group to direct the original packets from the input interfaces to the monitoring services interfaces for processing. The resulting flow description packets exit **fe-3/0/0** to reach the flow server.

Next, configure statements to port-mirror the monitored traffic to a tunnel interface. Design a firewall filter that selects some of this copied traffic for further analysis and some of the traffic for discarding. In this case, isolate TCP and UDP traffic and direct these two flows into separate filter-based forwarding routing instances. Remember to apply the filter to the tunnel interface to enable the separation of TCP traffic from UDP traffic. Also, import the interface routes into the forwarding instances with a routing table group.

In the filter-based forwarding instances, define static route next hops. The next hop for the TCP instance is the ES interface and the next hop for the UDP instance is the packet analyzer connected to **fe-3/2/0**. Finally, configure IPSec so that the next hop for the TCP traffic is the second packet analyzer attached to **fe-3/2/1**.

```
[edit]
interfaces {
   so-0/0/0 { # Traffic enters the router on this interface.
      description " input interface";
      encapsulation ppp;
      unit 0 {
         passive-monitor-mode; # Disables SONET keepalives.
         family inet {
            filter {
```

```
            input input-monitoring-filter; # The firewall filter is applied here.
          }
        }
      }
    }
    so-0/1/0 { # Traffic enters the router on this interface.
      description " input interface";
      encapsulation ppp;
      unit 0 {
        passive-monitor-mode;  # Disables SONET keepalives.
        family inet {
          filter {
            input input-monitoring-filter;  # The firewall filter is applied here.
          }
        }
      }
    }
    es-3/1/0 {  # This is where the TCP traffic enters the ES PIC.
      unit 0 {
        tunnel {
          source 10.8.8.1;
          destination 10.8.8.2;
        }
        family inet {
          ipsec-sa sa-esp;
          address 192.0.2.1/32 {
            destination 192.0.2.2;
          }
        }
      }
    }
    fe-3/0/0 { # Flow records exit here and travel to the flow server.
      description " export interface to the flow server";
      unit 0 {
        family inet;
        address 192.168.245.1/30;
      }
    }
    fe-3/2/0 { # This export interface for UDP traffic leads to a packet analyzer.
      description " export interface to the packet analyzer";
      unit 0 {
        family inet {
          address 10.9.9.1/30;
        }
```

```
    }
}
fe-3/2/1 {  # This IPSec tunnel source exports TCP traffic to a packet analyzer.
    unit 0 {
        family inet {
            address 10.8.8.1/30;
        }
    }
}
mo-4/0/0 { # This marks the beginning of the monitoring services interfaces.
    unit 0 { # Unit 0 is part of the inet.0 routing table and generates flow records.
        family inet;
    }
    unit 1 { # Unit 1 receives monitored traffic and is part of the VRF instance.
        family inet;
    }
}
mo-4/1/0 {
    unit 0 { # Unit 0 is part of the inet.0 routing table and generates flow records.
        family inet;
    }
    unit 1 { # Unit 1 receives monitored traffic and is part of the VRF instance.
        family inet;
    }
}
mo-4/2/0 {
    unit 0 { # Unit 0 is part of the inet.0 routing table and generates flow records.
        family inet;
    }
    unit 1 { # Unit 1 receives monitored traffic and is part of the VRF instance.
        family inet;
    }
}
mo-4/3/0 {
    unit 0 { # Unit 0 is part of the inet.0 routing table and generates flow records.
        family inet;
    }
    unit 1 { # Unit 1 receives monitored traffic and is part of the VRF instance.
        family inet;
    }
}
vt-0/2/0 {  # The tunnel services interface receives the port-mirrored traffic.
    unit 0 {
        family inet {
```

```
            filter {
               input tunnel-interface-filter;  # The filter splits traffic into TCP and UDP
            }
         }
      }
   }
}
forwarding-options {
   monitoring group1 {  # Monitored traffic is processed by the monitoring services
      family inet { # interfaces and flow records are sent to the flow server.
         output {
            export-format cflowd-version-5;
            flow-active-timeout 60;
            flow-inactive-timeout 30;
            flow-server 192.168.245.2 port 2055; # IP address and port for server.
            interface mo-4/0/0.1 {  # Use monitoring services interfaces for output.
               engine-id 1; # engine and interface-index statements are optional.
               engine-type 1;
               input-interface-index 44;
               output-interface-index 54;
               source-address 192.168.245.1; # This is the IP address of fe-3/0/0.
            }
            interface mo-4/1/0.1 {
               engine-id 2; # engine and interface-index statements are optional.
               engine-type 1;
               input-interface-index 45;
               output-interface-index 55;
               source-address 192.168.245.1; # This is the IP address of fe-3/0/0.
            }
            interface mo-4/2/0.1 {
               engine-id 3; # engine and interface-index statements are optional.
               engine-type 1;
               input-interface-index 46;
               output-interface-index 56;
               source-address 192.168.245.1; # This is the IP address of fe-3/0/0.
            }
            interface mo-4/3/0.1 {
               engine-id 4; # engine and interface-index statements are optional.
               engine-type 1;
               input-interface-index 47;
               output-interface-index 57;
               source-address 192.168.245.1; # This is the IP address of fe-3/0/0.
            }
         }
```

```
        }
    }
    port-mirroring {  # Copies the traffic and sends it to the Tunnel Services PIC.
        family inet {
            input {
                rate 1;
                run-length 1;
            }
            output {
                interface vt-0/2/0.0;
                no-filter-check;
            }
        }
    }
}
routing-options { # This installs the interface routes into the forwarding instances.
    interface-routes {
        rib-group inet bc-vrf;
    }
    rib-groups {
        bc-vrf {
            import-rib [inet.0 tcp-routing-table.inet.0 udp-routing-table.inet.0];
        }
    }
    forwarding-table {
        export pplb; # Applies per-packet load balancing to the forwarding table.
    }
}
policy-options {
    policy-statement monitoring-vrf-import {
        then reject;
    }
    policy-statement monitoring-vrf-export {
        then reject;
    }
    policy-statement pplb {
        then {
            load-balance per-packet;
        }
    }
}
security { # This sets IPSec options for the ES PIC.
    ipsec {
        proposal esp-sha1-3des {
```

```
            protocol esp;
            authentication-algorithm hmac-sha1-96;
            encryption-algorithm 3des-cbc;
            lifetime-seconds 180;
        }
        policy esp-group2 {
            perfect-forward-secrecy {
                keys group2;
            }
            proposals esp-sha1-3des;
        }
        security-association sa-esp {
            mode tunnel;
            dynamic {
                ipsec-policy esp-group2;
            }
        }
    }
    ike {
        proposal ike-esp {
            authentication-method pre-shared-keys;
            dh-group group2;
            authentication-algorithm sha1;
            encryption-algorithm 3des-cbc;
            lifetime-seconds 180;
        }
        policy 10.8.8.2 {
            mode aggressive;
            proposals ike-esp;
            pre-shared-key ascii-text "$ABC123";
        }
    }
}
firewall {
    family inet {
        filter input-monitoring-filter { # This filter selects traffic to send into the VRF
            term 1 { # instance and prepares the traffic for port mirroring.
                from {
                    destination-address {
                        10.7.0.0/16;
                    }
                }
                then {
                    port-mirror;
```

```
            accept;
          }
        }
        term 2 {
          from {
            destination-address {
              10.6.0.0/16;
            }
          }
          then accept;
        }
      }
      filter tunnel-interface-filter {  # This filter breaks the port-mirrored traffic into two
        term tcp { # filter-based forwarding instances: TCP packets and UDP packets.
          from {
            protocol tcp;
          }
          then { # This counts TCP packets and sends them into a TCP instance.
            count tcp;
            routing-instance tcp-routing-table;
          }
        }
        term udp {
          from {
            protocol udp;
          }
          then { # This counts UDP packets and sends them into a UDP instance.
            count udp;
            routing-instance udp-routing-table;
          }
        }
        term rest {
          then {
            count rest;
            discard;
          }
        }
      }
    }
}
routing-instances {
    monitoring-vrf { # This is the VRF instance where you send the traffic. It contains
      instance-type vrf; # the input interface and the monitoring services interfaces.
      interface so-0/0/0.0; # Traffic enters the router on these input interfaces.
```

```
        interface so-0/1/0.0;
        interface mo-4/0/0.1;
        interface mo-4/1/0.1; # These are output interfaces (use them as
        interface mo-4/2/0.1; # output interfaces in your monitoring group).
        interface mo-4/3/0.1;
        route-distinguisher 69:1;
        vrf-import monitoring-vrf-import;
        vrf-export monitoring-vrf-export;
        routing-options { # Sends traffic to a group of monitoring services interfaces.
            static {
                route 0.0.0.0/0 next-hop [mo-4/0/0.1 mo-4/1/0.1
                    mo-4/2/0.1 mo-4/3/0.1];
            }
        }
    }
    tcp-routing-table {  # This is the filter-based forwarding instance for TCP traffic.
        instance-type forwarding;
        routing-options {  # The next hop is the ES PIC.
            static {
                route 0.0.0.0/0 next-hop es-3/1/0.0;
            }
        }
    }
    udp-routing-table { # This is the filter-based forwarding instance for UDP traffic.
        instance-type forwarding;
        routing-options { # The next hop is the second packet analyzer.
            static {
                route 0.0.0.0/0 next-hop 10.9.1.2;
            }
        }
    }
}
```

Verifying Your Work

To verify that your configuration is correct, use the following commands on the monitoring station that is configured for passive flow monitoring:

- **show route 0/0**
- **show passive-monitoring error**
- **show passive-monitoring flow**
- **show passive-monitoring memory**

- **show passive-monitoring status**
- **show passive-monitoring usage**

To clear statistics for the **show passive-monitoring error** and **show passive-monitoring flow** commands, issue the **clear passive-monitoring (all | *interface-name*)** command.

You can also view passive flow monitoring status with the Simple Network Management Protocol (SNMP). The following Management Information Base (MIB) tables are supported:

- **jnxPMonErrorTable**—Corresponds to the **show passive-monitoring error** command.
- **jnxPMonFlowTable**—Corresponds to the **show passive-monitoring flow** command.
- **jnxPMonMemoryTable**—Corresponds to the **show passive-monitoring memory** command.

The following section shows the output of the **show** commands used with the configuration example:

```
user@host> show route 0/0
```

```
 <skip inet.0>
```

We are only concerned with the routing-instance route.

```
bc-vrf.inet.0:  1 destinations, 1 routes (1 active, 0 holddown, 0 hidden)
bc-vrf.inet.0:+ = Active Route, - = Last Active, * = Both
0.0.0.0/0       *[Static/5] 5d 17:34:57
                 via mo-4/0/0.1
                > via mo-4/1/0.1
                 via mo-4/2/0.1
                 via mo-4/3/0.1
tcp-rt.inet.0: 13 destinations, 13 routes (12 active, 0 holddown, 1
hidden)
+ = Active Route, - = Last Active, * = Both
0.0.0.0/0          *[Static/5] 19:24:39
                    > via es-3/1/0.0
  : <other interface routes>
udp-rt.inet.0: 13 destinations, 13 routes (12 active, 0 holddown, 1
hidden)
+ = Active Route, - = Last Active, * = Both
0.0.0.0/0          *[Static/5] 19:24:39
                    > to 10.9.1.2 via fe-3/2/0.0
  : <other interface routes>
```

NOTE: For all **show passive-monitoring** commands, the output obtained when using a wildcard (such as *) or the **all** option is based on the configured interfaces listed at the **[edit forwarding-options monitoring *group-name*]** hierarchy level. In the output from the configuration example, you see information only for the configured interfaces **mo-4/0/0**, **mo-4/1/0**, **mo-4/2/0**, and **mo-4/3/0**.

Many of the statements you can configure in a monitoring group, such as **engine-id** and **engine-type**, are visible in the output of the **show passive-monitoring** commands.

Table 29: Output Fields for the show passive-monitoring error Command

Field	Explanation
Packets dropped (no memory)	Number of packets dropped because of memory.
Packets dropped (not IP)	Number of non-IP packets dropped.
Packets dropped (not IPv4)	Number of packets dropped because they failed the IPv4 check.
Packets dropped (header too small)	Number of packets dropped because the packet length or IP header length was too small.
Memory allocation failures	Number of flow record memory allocation failures. A small number reflects failures to replenish the free list. A large number indicates the monitoring station is almost out of memory space.
Memory free failures	Number of flow record memory frees.
Memory free list failures	Number of flow records received from free list that failed. Memory is nearly exhausted or too many new flows greater than 128K are being created in one second.
Memory warning	The flows have exceeded 1 million packets per second (Mpps) on a Monitoring Services PIC or 2 Mpps on a Monitoring Services II PIC. The response can be **Yes** or **No**.
Memory overload	The memory has been overloaded. The response is **Yes** or **No**.

Table 29: Output Fields for the show passive-monitoring error Command *(continued)*

Field	Explanation
PPS overload	In packets per second, whether the PIC is receiving more traffic than the configured threshold. The response can be **Yes** or **No**.
BPS overload	In bytes per second, whether the PIC is receiving more traffic than the configured threshold. The response can be **Yes** or **No**.

`user@host>` **show passive-monitoring error all**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory warning: No, Memory overload: No, PPS overload: No, BPS overload: No

Passive monitoring interface: mo-4/1/0, Local interface index: 45
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory warning: No, Memory overload: No, PPS overload: No, BPS overload: No

Passive monitoring interface: mo-4/2/0, Local interface index: 46
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory warning: No, Memory overload: No, PPS overload: No, BPS overload: No

Passive monitoring interface: mo-4/3/0, Local interface index: 47
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
```

```
    Memory free list failures: 0
    Memory warning: No, Memory overload: No, PPS overload: No, BPS overload: No
```

Table 30: Output Fields for the show passive-monitoring flow Command

Field	Explanation
Flow packets	Number of packets received by an operational PIC.
Flow bytes	Number of bytes received by an operational PIC.
Flow packets 10-second rate	Number of packets per second handled by the PIC and displayed as a 10-second average.
Flow bytes 10-second rate	Number of bytes per second handled by the PIC and displayed as a 10-second average.
Active flows	Number of currently active flows tracked by the PIC.
Total flows	Total number of flows received by an operational PIC.
Flows exported	Total number of flows exported by an operational PIC.
Flows packets exported	Total number of flow packets exported by an operational PIC.
Flows inactive timed out	Total number of flows that are exported because of inactivity.
Flows active timed out	Total number of long-lived flows that are exported because of an active timeout.

`user@host>` **show passive-monitoring flow all**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  Flow information
```

```
    Flow packets: 6533434, Flow bytes: 653343400
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 1599
    Flows exported: 1599, Flows packets exported: 55
    Flows inactive timed out: 1599, Flows active timed out: 0

Passive monitoring interface: mo-4/1/0, Local interface index: 45
  Flow information
    Flow packets: 6537780, Flow bytes: 653778000
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 1601
    Flows exported: 1601, Flows packets exported: 55
    Flows inactive timed out: 1601, Flows active timed out: 0

Passive monitoring interface: mo-4/2/0, Local interface index: 46
  Flow information
    Flow packets: 6529259, Flow bytes: 652925900
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 1599
    Flows exported: 1599, Flows packets exported: 55
    Flows inactive timed out: 1599, Flows active timed out: 0

Passive monitoring interface: mo-4/3/0, Local interface index: 47
  Flow information
    Flow packets: 6560741, Flow bytes: 656074100
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 1598
    Flows exported: 1598, Flows packets exported: 55
    Flows inactive timed out: 1598, Flows active timed out: 0
```

Table 31: Output Fields for the show passive-monitoring memory Command

Field	Explanation
Allocation count	Number of flow records allocated.
Free count	Number of flow records freed.
Maximum allocated	Maximum number of flow records allocated since the monitoring station booted. This number represents the peak number of flow records allocated at a time.

Table 31: Output Fields for the show passive-monitoring memory Command *(continued)*

Field	Explanation
Allocations per second	Flow records allocated per second during the last statistics interval on the PIC.
Frees per second	Flow records freed per second during the last statistics interval on the PIC.
Total memory used	Total amount of memory currently used (in bytes).
Total memory free	Total amount of memory currently free (in bytes).

`user@host>` **show passive-monitoring memory all**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  Memory utilization
    Allocation count: 1600, Free count: 1599, Maximum allocated: 1600
    Allocations per second: 3200, Frees per second: 1438
   Total memory used (in bytes): 103579176, Total memory free (in bytes): 163914184

Passive monitoring interface: mo-4/1/0, Local interface index: 45
  Memory utilization
    Allocation count: 1602, Free count: 1601, Maximum allocated: 1602
    Allocations per second: 3204, Frees per second: 1472
   Total memory used (in bytes): 103579176, Total memory free (in bytes): 163914184

Passive monitoring interface: mo-4/2/0, Local interface index: 46
  Memory utilization
    Allocation count: 1600, Free count: 1599, Maximum allocated: 1600
    Allocations per second: 3200, Frees per second: 1440
   Total memory used (in bytes): 103579176, Total memory free (in bytes): 163914184

Passive monitoring interface: mo-4/3/0, Local interface index: 47
  Memory utilization
    Allocation count: 1599, Free count: 1598, Maximum allocated: 1599
    Allocations per second: 3198, Frees per second: 1468
   Total memory used (in bytes): 103579176, Total memory free (in bytes): 163914184
```

Table 32: Output Fields for the show passive-monitoring status Command

Field	Explanation
Interface state	Indicates whether the interface is monitoring (operating properly), disabled (administratively disabled), or not monitoring (not configured).
Group index	Integer that represents the monitoring group of which the PIC is a member. (This does not indicate the number of monitoring groups.)
Export interval	Configured export interval for flow records, in seconds.
Export format	Configured export format (only v5 is currently supported).
Protocol	Protocol the PIC is configured to monitor (only IPv4 is currently supported).
Engine type	Configured engine type that is inserted in output flow packets.
Engine ID	Configured engine ID that is inserted in output flow packets.
Route record count	Number of routes recorded.
IFL to SNMP index count	Number of logical interfaces mapped to an SNMP index.
AS count	Number of AS boundaries that the flow has crossed.
Time set	Indicates whether the time stamp is in place.
Configuration set	Indicates whether the monitoring configuration is set.
Route record set	Indicates whether routes are being recorded.
IFL SNMP map set	Indicates whether logical interfaces are being mapped to an SNMP index.

`user@host>` **show passive-monitoring status all**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  Interface state: Monitoring
  Group index: 0
  Export interval: 15 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 1, Engine ID: 1
  Route record count: 13, IFL to SNMP index count: 30, AS count: 1
  Time set: Yes, Configuration set: Yes
```

```
  Route record set: Yes, IFL SNMP map set: Yes

Passive monitoring interface: mo-4/1/0, Local interface index: 45
  Interface state: Monitoring
  Group index: 0
  Export interval: 15 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 1, Engine ID: 2
  Route record count: 13, IFL to SNMP index count: 30, AS count: 1
  Time set: Yes, Configuration set: Yes
  Route record set: Yes, IFL SNMP map set: Yes

Passive monitoring interface: mo-4/2/0, Local interface index: 46
  Interface state: Monitoring
  Group index: 0
  Export interval: 15 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 1, Engine ID: 3
  Route record count: 13, IFL to SNMP index count: 30, AS count: 1
  Time set: Yes, Configuration set: Yes
  Route record set: Yes, IFL SNMP map set: Yes

Passive monitoring interface: mo-4/3/0, Local interface index: 47
  Interface state: Monitoring
  Group index: 0
  Export interval: 15 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 1, Engine ID: 4
  Route record count: 13, IFL to SNMP index count: 30, AS count: 1
  Time set: Yes, Configuration set: Yes
  Route record set: Yes, IFL SNMP map set: Yes
```

Table 33: Output Fields for the show passive-monitoring usage Command

Field	Explanation
Uptime	Time, in milliseconds, that the PIC has been operational.
Interrupt time	Cumulative time that the PIC spent in processing packets since the last PIC reset.
Load (5 second)	CPU load on the PIC averaged over 5 seconds. The number is a percentage obtained by dividing the time spent on active tasks by the total elapsed time.

Table 33: Output Fields for the show passive-monitoring usage Command *(continued)*

Field	Explanation
Load (1 minute)	CPU load on the PIC averaged over 1 minute. The number is a percentage obtained by dividing the time spent on active tasks by the total elapsed time.

`user@host>` **show passive-monitoring usage** *

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  CPU utilization
    Uptime: 653155 milliseconds, Interrupt time: 40213754 microseconds
    Load (5 second): 20%, Load (1 minute): 17%

Passive monitoring interface: mo-4/1/0, Local interface index: 45
  CPU utilization
    Uptime: 652292 milliseconds, Interrupt time: 40223178 microseconds
    Load (5 second): 22%, Load (1 minute): 15%

Passive monitoring interface: mo-4/2/0, Local interface index: 46
  CPU utilization
    Uptime: 649491 milliseconds, Interrupt time: 40173645 microseconds
    Load (5 second): 22%, Load (1 minute): 10098862%

Passive monitoring interface: mo-4/3/0, Local interface index: 47
  CPU utilization
    Uptime: 657328 milliseconds, Interrupt time: 40368704 microseconds
    Load (5 second): 1%, Load (1 minute): 15%
```

Copying and Redirecting Traffic with Port Mirroring and Filter-Based Forwarding

This section discusses additional techniques you can use with the passive flow monitoring application:

- In addition to flow analysis, you can analyze a copy of the original traffic with a single packet analyzer. To implement this technique, divert traffic with a filter-based forwarding routing instance and send the monitored traffic through a physical interface to the packet analyzer.

- You can cluster the traffic into different groups and redirect this traffic to multiple packet analyzers. For example, you can break traffic flows into TCP groups and UDP groups and send these groups of packets to different analyzers. To accomplish this, you use port mirroring and send a copy of the original traffic to a Tunnel PIC. Then you can apply a firewall filter, split the traffic into your desired groups, and send these groups toward different exit interfaces leading to the packet analyzers. This technique provides maximum flexibility for traffic analysis.
- For secure transmission of the copied or grouped traffic, you can encrypt the diverted traffic with an ES PIC and send this traffic to a packet analyzer over an IP Security (IPSec) tunnel.

To implement the filter-based forwarding enhancement methods, see the following sections:

Specifying Port Mirroring Input and Output on M, MX or T Series Routers

This step works in conjunction with the action specified by the **port-mirror** statement configured at the **[edit firewall family (inet | inet6) filter *filter-name* term *term-name* then]** hierarchy level. At this point, you select input and output statements to determine where the copies of the IPv4 or IPv6 packets are sent. To configure, include the **input** and **output** statements at the **[edit forwarding-options port-mirroring family *family-name*]** hierarchy level. The traffic to be monitored is copied, port-mirrored, and sent to the packet analyzer for analysis.

> **NOTE:** On M Series routers, you can port-mirror either IPv4 or IPv6 packets at one time. On M120, M320, and T Series routers, you can port-mirror both IPv4 and IPv6 packets simultaneously.

NOTE: On an M320 or T Series router using an Adaptive Services (AS) II PIC or a MultiServies PIC, corrupted IP packets might be sent to the port mirror when traffic passes through an IPSec tunnel. The inbound IP traffic passes through the IPSec tunnel and the **sp** interface is decoded and forwarded to the port mirror correctly, but the return outbound traffic is corrupted and unreadable through the router configured with the port mirror.

The port-mirrored copy of the traffic can travel only to a single next hop. As a result, only one type of analysis can be performed if the packets are sent to a packet analyzer through a physical next hop. If more than one type of analysis is desired, a tunnel interface must be used as the next hop for port mirroring. When the mirrored copy of the traffic arrives at the virtual tunnel interface, it can be filtered, split into groups, and redirected to multiple exit interfaces and packet analyzers.

For your input requirements, include the **rate** and **run-length** statements at the **[edit forwarding-options port-mirroring family *family-name* input]** hierarchy level. For your output requirements, specify the target interface with the **interface** statement at the **[edit forwarding-options port-mirroring family *family-name* output]** hierarchy level.

By default, a filter cannot be applied to an interface where port-mirrored traffic is received. To allow the tunnel services interface to be used as a filtered next hop, include the **no-filter-check** statement at the **[edit forwarding-options port-mirroring family *family-name* output]** hierarchy level.

```
[edit]
forwarding-options {
    port-mirroring {
        family (inet | inet6) {
            input {
                rate 1;
                run-length 5;
            }
            output {
                interface vt-0/2/0.0;
                no-filter-check;
            }
        }
    }
}
```

NOTE: Before Junos OS Release 7.4, you could configure the **input** and **output** statements at the **[edit forwarding-options port-mirroring]** hierarchy level. However, this older syntax has been revised to extend port-mirroring support to IPv6 packets. If you have a configuration that contains the older syntax, we recommend that you update your configuration to the new syntax listed above.

Creating a Firewall Filter on an M, MX or T Series Router to Split the Port-Mirrored Traffic into Different Instances

If you need to split the copy of the monitored traffic into separate groups and send these filtered packets to different analyzers, devise a firewall filter that selects some traffic for sampling and some traffic for discarding. In this case, UDP traffic is sent into one routing instance, TCP traffic is diverted into a second routing instance, and all other traffic is discarded. In a later step, you will define the filter-based forwarding routing instances specified in the **then** statements shown in this filter.

```
[edit]
firewall {
    family inet {
        filter tunnel-interface-filter {
            term tcp {
                from {
                    protocol tcp;
                }
                then {
                    count tcp;
                    routing-instance tcp-routing-table;
                }
            }
            term udp {
                from {
                    protocol udp;
                }
                then {
                    count udp;
                    routing-instance udp-routing-table;
                }
            }
            term rest {
```

```
            then {
                count rest;
                discard;
            }
        }
    }
}
```

Applying the Firewall Filter to a Tunnel PIC Interface

Once the firewall filter is defined, apply it as an input filter on a tunnel interface. This is required if the firewall filter defines two or more types of traffic or export interfaces. However, if the firewall filter only specifies one type of traffic and one export interface, you can apply the filter directly to the export interface.

```
[edit]
interfaces {
    vt-0/2/0 {
        unit 0 {
            family inet {
                filter {
                    input tunnel-interface-filter;
                }
            }
        }
    }
}
```

Using Filter-Based Forwarding to Export Monitored Traffic to Multiple Destinations

The firewall filter called **tunnel-interface-filter** sends UDP traffic into one filter-based forwarding routing instance called **udp-routing-table**, sends TCP traffic into a second filter-based forwarding routing instance called **tcp-routing-table**, and discards all other packets. Here you will configure the filter-based forwarding instances.

Configure an export interface for each of your routing instances by including a static next hop. To configure, include the **route** statement at the **[edit routing-instances *instance-name* routing-options static]** hierarchy level and specify a next-hop address or interface.

```
[edit]
```

```
routing-instances {
    tcp-routing-table {
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop es-3/1/0.0;
            }
        }
    }
    udp-routing-table {
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop 10.9.1.2;
            }
        }
    }
}
```

Configuring a Routing Table Group on an M, MX or T Series Router to Add Interface Routes into the Forwarding Instance

Next, import the interface routes into the forwarding instance. This step is necessary because the next hops specified in the forwarding instances must be installed in the forwarding instances themselves. To configure, include the **import-rib** statement at the **[edit routing-options rib-groups *group-name*]** hierarchy level. The **export** statement at the **[edit routing-options forwarding-table]** hierarchy level and the **pplb** policy enable load balancing.

```
[edit]
routing-options {
    interface-routes {
        rib-group inet bc-vrf;
    }
    rib-groups {
        bc-vrf {
            import-rib [inet.0 tcp-routing-table.inet.0 udp-routing-table.inet.0];
        }
    }
    forwarding-table {
        export pplb;
```

```
    }
}
policy-options {
    policy-statement pplb {
        then {
            load-balance per-packet;
        }
    }
}
```

Using IPSec and an ES PIC on an M, MX or T Series Router to Send Encrypted Traffic to a Packet Analyzer

You can send some or all of the traffic securely to the packet analyzer using IPSec (a suite of related protocols for cryptographically securing communications at the IP Packet Layer) and an Encryption Services (ES) PIC. In this case, the TCP traffic is encrypted, sent over an IPSec tunnel, and received by the packet analyzer. For more information on configuring IPSec on the ES PIC, see the *IPsec User Guide* or the *Junos System Basics Configuration Guide*.

```
[edit]
interfaces {
    es-3/1/0 {
        unit 0 {
            tunnel {
                source 10.8.8.1;
                destination 10.8.8.2;
            }
            family inet {
                ipsec-sa sa-esp;
                address 192.0.2.1/32 {
                    destination 192.0.2.2;
                }
            }
        }
    }
    fe-3/2/1 {
        unit 0 {
            family inet {
                address 10.8.8.1/30;
            }
```

```
        }
    }
}
security {
    ipsec {
        proposal esp-sha1-3des {
            protocol esp;
            authentication-algorithm hmac-sha1-96;
            encryption-algorithm 3des-cbc;
            lifetime-seconds 180;
        }
        policy esp-group2 {
            perfect-forward-secrecy {
                keys group2;
            }
            proposals esp-sha1-3des;
        }
        security-association sa-esp {
            mode tunnel;
            dynamic {
                ipsec-policy esp-group2;
            }
        }
    }
    ike {
        proposal ike-esp {
            authentication-method pre-shared-keys;
            dh-group group2;
            authentication-algorithm sha1;
            encryption-algorithm 3des-cbc;
            lifetime-seconds 180;
        }
        policy 10.8.8.2 {
            mode aggressive;
            proposals ike-esp;
            pre-shared-key ascii-text "$ABC123";
        }
    }
}
```

Applying a Firewall Filter Output Interface on an M, MX or T Series Router to Port-mirror Traffic to PICs or Flow Collection Services

On output interfaces, you can apply a firewall filter that leads to a filter-based forwarding routing instance. This is useful if you want to port-mirror traffic to multiple Monitoring Services PICs or flow collection services interfaces. To configure, include the **output** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet filter]** hierarchy level.

```
[edit]
interfaces
fe-3/1/0 {
    description “export interface to flow collection services interfaces”;
    unit 0 {
        family inet;
        address ip-address;
        filter {
            output output-filter-name;
        }
    }
}
```

Monitoring Traffic on a Router with a VRF Instance and a Monitoring Group

The first way you can implement passive flow monitoring is to direct traffic into a VRF routing instance and use a monitoring group to export this traffic to a flow server for analysis. Complete the following tasks:

- Specifying a Firewall Filter on an M, MX or T Series Router to Select Traffic to Monitor on page 238
- *Configuring Input Interfaces, Monitoring Services Interfaces and Export Interfaces on M, MX or T Series Routers*
- Establishing a VRF Instance on an M, MX or T Series Router for Monitored Traffic on page 242
- *Configuring a Monitoring Group on an M, MX or T Series Router to Send Traffic to the Flow Server*
- *Configuring Policy Options on M, MX or T Series Routers*
- *Stripping MPLS Labels on ATM, Ethernet-Based and SONET/SDH Router Interfaces*

Specifying a Firewall Filter on an M, MX or T Series Router to Select Traffic to Monitor

When you define a firewall filter, you select the initial traffic to be monitored. To configure a firewall filter, include the **filter** statement at the **[edit firewall family inet]** hierarchy level. All filtered traffic to be monitored must be accepted.

```
[edit]
firewall {
    family inet {
        filter input-monitoring-filter {
            term 1 {
                from {
                    destination-address {
                        10.7.0.0/16;
                    }
                }
                then {
                    count counter1;
                    accept;
                }
            }
            term 2 {
                from {
                    destination-address {
                        10.6.0.0/16;
                    }
                }
                then {
                    count counter2;
                    accept;
                }
            }
        }
    }
}
```

Configuring Input Interfaces, Monitoring Services Interfaces and Export Interfaces on M, MX or T Series Routers

After creating the input filter, you need to configure the interfaces where traffic will enter the router. To enable passive flow monitoring for SONET/SDH input interfaces, include the **passive-monitor-mode** statement at the **[edit interfaces so-*fpc/pic/port* unit *unit-number*]** hierarchy level. This mode disables the router from participating in the network as an active device. On SONET/SDH interfaces, passive monitor mode suppresses SONET keepalives.

For ATM2 IQ interfaces, passive monitor mode suppresses the sending and receiving of ATM Operations, Administration, and Maintenance (OAM) and Integrated Local Management Interface (ILMI) control messages. To enable passive flow monitoring for ATM2 IQ input interfaces, include the **passive-monitor-mode** statement at the **[edit interfaces at-*fpc/pic/port*]** hierarchy level. ATM passive monitoring supports the following interface encapsulation types: Cisco-compatible ATM Network Layer Protocol ID (NLPID) (**atm-cisco-nlpid**), ATM NLPID (**atm-nlpid**), ATM Point-to-Point Protocol (PPP) over ATM Adaptation Layer 5 (AAL5)/ logical link control (LLC) (**atm-ppp-llc**), ATM PPP over raw AAL5 (**atm-ppp-vc-mux**), ATM LLC/ subnetwork attachment point (SNAP) (**atm-snap**), and ATM virtual circuit (VC) multiplexing (**atm-vc-mux**).

Ethernet-based interfaces support both per-port passive monitoring and per-VLAN passive monitoring. For Fast Ethernet interfaces, include the **passive-monitor-mode** statement at the **[edit interfaces fe-*fpc/pic/port*]** hierarchy level. For Gigabit Ethernet interfaces, include the **passive-monitor-mode** statement at the **[edit interfaces ge-*fpc/pic/port*]** hierarchy level. On Ethernet-based interfaces, passive monitor mode disables the Routing Engine from receiving packets and prevents the routing table from transmitting packets. You can verify this by the presence of the **No-receive** and **No-transmit** interface flags in the output of the **show interfaces (fe | ge)-*fpc/pic/port*** command.

NOTE: The following restrictions apply to passive flow monitoring on Ethernet-based interfaces:

- No special encapsulation types are allowed, so you must configure Ethernet encapsulations only.
- When you configure the **passive-monitor-mode** statement, destination MAC address filters applied to incoming interfaces are disabled by default.
- The **flow-control** statement at the **[edit interfaces ge-*fpc/pic/port* gigether-options**] or **[edit interfaces fe-*fpc/pic/port* fastether-options**] hierarchy level does not work when passive flow monitoring is enabled.

In addition to passive monitor mode, apply the previously defined firewall filter to the interface with the **filter** statement at the **[edit interfaces *interface-name-fpc/pic/port* unit *unit-number* family inet]** hierarchy level:

```
[edit]
interfaces {
    so-0/0/0 {
        description “SONET/SDH input interface”;
        encapsulation ppp;
        unit 0 {
            passive-monitor-mode;
            family inet {
                filter {
                    input  input-monitoring-filter;
                }
            }
        }
    }
    at-1/0/0 {
        description “ATM2 IQ input interface”;
        passive-monitor-mode;
        atm-options {
            pic-type atm2;
            vpi 0 {
                maximum-vcs 255;
            }
        }
        unit 0 {
            encapsulation atm-snap;
            vci 0.100;
            family inet {
                filter {
                    input input-monitoring-filter;
                }
            }
        }
    }
    ge-2/0/0 {
        description “Gigabit Ethernet input interface”;
        passive-monitor-mode;
        unit 0 {
            family inet {
                filter {
                    input input-monitoring-filter;
                }
            }
        }
    }
```

```
}
```

Configure the interfaces on the Monitoring Services PIC or Monitoring Services II PIC with the **family inet** statement at the **[edit interfaces mo-*fpc/pic/port* unit *unit-number*]** hierarchy level. The statement allows the interfaces to process IPv4 traffic received from the input interfaces.

When you use VRF instances, you need to configure two logical interfaces. The first (**unit 0**) is part of the inet.0 routing table and sources the flow packets. The second (**unit 1**) is configured as part of the VRF instance so the monitoring services interface can serve as a valid next hop for packets received in the instance.

You can also capture options packets and time-to-live (TTL) exceeded information when the monitoring services interface processes flow records. To configure, include the **receive-options-packets** and **receive-ttl-exceeded** statements at the **[edit interfaces mo-*fpc/pic/port* unit *unit-number* family inet]** hierarchy level:

```
[edit]
interfaces {
    mo-4/0/0 {
        unit 0 {
            family inet {
                receive-options-packets;
                receive-ttl-exceeded;
            }
        }
        unit 1 {
            family inet;
        }
    }
    mo-4/1/0 {
        unit 0 {
            family inet;
        }
        unit 1 {
            family inet;
        }
    }
    mo-4/2/0 {
        unit 0 {
            family inet;
        }
        unit 1 {
            family inet;
        }
```

```
    }
    mo-4/3/0 {
        unit 0 {
            family inet;
        }
        unit 1 {
            family inet;
        }
    }
}
```

You must also configure the export interface where flow packets exit the monitoring station and are sent to the flow server.

On output interfaces, you can apply a firewall filter that leads to a filter-based forwarding routing instance. This is useful if you want to port-mirror traffic to multiple Monitoring Services PICs or flow collection services interfaces. To configure, include the **output** statement at the **[edit interfaces *interface-name* unit *logical-unit-number* family inet filter]** hierarchy level. For more information, see "Using Filter-Based Forwarding to Export Monitored Traffic to Multiple Destinations" on page 233.

```
[edit]
interfaces
fe-3/0/0 {
    description "export interface to flow server";
    unit 0 {
        family inet;
        address ip-address;
        filter {
            output output-filter-name;
        }
    }
}
```

Establishing a VRF Instance on an M, MX or T Series Router for Monitored Traffic

After the firewall filter and interfaces are ready, create a VPN routing and forwarding (VRF) instance. The filtered traffic enters the VRF instance and is shared only between the input interfaces and the monitoring services output interfaces. In this case, a group of four monitoring services interfaces is used as the next hop.

```
[edit]
routing-instances {
   monitoring-vrf {
      instance-type vrf;
      interface so-0/0/0.0;
      interface so-0/1/0.0;
      interface mo-4/0/0.1;
      interface mo-4/1/0.1;
      interface mo-4/2/0.1;
      route-distinguisher 69:1;
      vrf-import monitoring-vrf-import;
      vrf-export monitoring-vrf-export;
      routing-options {
         static {
            route 0.0.0.0/0 next-hop [mo-4/0/0.1 mo-4/1/0.1 mo-4/2/0.1];
         }
      }
   }
}
```

Configuring a Monitoring Group on an M, MX or T Series Router to Send Traffic to the Flow Server

You collect flow records by specifying output interfaces in a monitoring group. In general, the monitoring services interfaces are the output interfaces. The logical unit number on the output interfaces when used in conjunction with a VRF instance must be 1. To configure, include the **output** statement at the **[edit forwarding-options monitoring *group-name* family inet]** hierarchy level.

> **NOTE:** Because routing instances determine the input interface, the **input** statement at the **[edit forwarding-options monitoring *group-name* family inet]** hierarchy level has been removed in Junos OS Release 6.0 and later. If you have a configuration that contains this old statement, we recommend that you update your configuration and remove the statement.

As part of the **mo-*fpc*/*pic*/*port*** statement at the **[edit forwarding-options monitoring *group-name* family inet output interface]** hierarchy level, you must specify a source address for transmission of flow information. You can use the router ID IP address, the IP address of the input interface, or any local IP address of your choice as the source address. If you provide a different **source-address** statement for each monitoring services output interface, you can track which interface processes a particular flow record.

All other statements at this level (**engine-id, engine-type, input-interface-index,** and **output-interface-index**) are dynamically generated, but can be configured manually. To reset outgoing interface or incoming interface indexes that were once configured manually, configure the **input-interface-index** or **outgoing-interface-index** statements with a value of 0 at the **[edit forwarding-options monitoring *group-name* family inet output interface *interface-name*]** hierarchy level.

To specify the flow server IP address and port number, include the **flow-server *ip-address* port *port-number*** statement at the **[edit forwarding-options monitoring *group-name* family inet output]** hierarchy level. You can specify up to eight flow servers in a monitoring group and the IP address for each server must be unique. Flow records are exported and load-balanced between all active flow servers.

Once you configure the VRF and monitoring group statements, traffic enters the input interfaces, passes to the monitoring services interfaces for processing, and is discarded. The resulting flow description packets exit the monitoring station through the export interface.If you want traffic to travel to destinations other than the monitoring services interfaces, or need to establish additional analysis, see the section "Copying and Redirecting Traffic with Port Mirroring and Filter-Based Forwarding" on page 229.

NOTE: You must complete interface configuration on the Monitoring Services or Monitoring Services II PIC before an interface can be added into a monitoring group. For more information, see "Configuring Input Interfaces, Monitoring Services Interfaces and Export Interfaces on M, MX or T Series Routers" on page 239.

```
[edit]
forwarding-options {
    monitoring group1 {
        family inet {
            output {
                export-format cflowd-version-5;
                flow-active-timeout 60;
                flow-inactive-timeout 30;
                flow-server 192.168.245.1 port 2055;
                flow-server 192.168.245.2 port 2055;
                interface mo-4/0/0.1 {
                    engine-id 1;
                    engine-type 1;
                    input-interface-index 44;
                    output-interface-index 54;
                    source-address 192.168.245.1;
                }
                interface mo-4/1/0.1 {
                    engine-id 2;
                    engine-type 1;
```

```
            input-interface-index 45;
            output-interface-index 55;
            source-address 192.168.245.1;
        }
        interface mo-4/2/0.1 {
            engine-id 3;
            engine-type 1;
            input-interface-index 46;
            output-interface-index 56;
            source-address 192.168.245.1;
        }
    }
  }
}
```

Configuring Policy Options on M, MX or T Series Routers

When you use a group of next hops in your monitoring group, you can load-balance traffic and distribute it to the export interfaces if you configure policy options. To configure, include the **load-balance per-packet** statement at the **[edit policy-options policy-statement *policy-name* then]** hierarchy level. You can also reject import and export of VRF routes by including the **reject** statement at the **[edit policy-options policy-statement *policy-name* then]** hierarchy level.

```
[edit]
routing-options {
  forwarding-table {
    export pplb;
  }
}
policy-options {
  policy-statement monitoring-vrf-import {
    then {
      reject;
    }
  }
  policy-statement monitoring-vrf-export {
    then {
      reject;
    }
  }
```

```
    policy-statement pplb {
        then {
            load-balance per-packet;
        }
    }
}
```

Stripping MPLS Labels on ATM, Ethernet-Based and SONET/SDH Router Interfaces

Because flow monitoring can be performed only on IPv4 packets, any packets containing MPLS labels must have the labels removed before monitoring can occur. To remove MPLS labels from packets as they enter an ATM2 IQ, Ethernet-based, or SONET/SDH interface, include the **pop-all-labels** statement at the **[edit interfaces *interface-name-fpc/pic/port* (atm | fastether | gigether | sonet)-options mpls]** hierarchy level. If you use static MPLS labels, we recommend you assign label values from **10000** through **99999** to avoid using the label ranges reserved by the Junos OS.

To remove a specified number of labels from selected packets with MPLS labels, include the **required-depth** statement at the **[edit interfaces *interface-name-fpc/pic/port* (atm | fastether | gigether | sonet)-options mpls pop-all-labels]** hierarchy level. A **required-depth** value of **1** removes labels from all packets containing only one MPLS label, a value of **2** removes labels from all packets containing only two MPLS labels, and a value of **[1 2]** removes labels from all packets containing either one or two MPLS labels. The **required-depth** value of **[1 2]** is the default setting. When you configure the **required-depth** statement, you must configure the same value for all ports on the same PIC.

The labels are removed and discarded as soon as they arrive at the interface. As a result, no MPLS filters can be applied to the stripped labels, no statistics are generated for the labels, and you cannot apply an IP filter to the incoming packets. No Tunnel Services PIC is required to perform MPLS label stripping.

```
[edit]
interfaces {
    at-/fpc/pic/port  {
        atm-options {
            mpls {
                pop-all-labels {
                    required-depth 1;
                }
            }
        }
    }
```

```
    (fe | ge)-fpc/pic/port {
        (fastether | gigether)-options {
            mpls {
                pop-all-labels {
                    required-depth [1 2];
                }
            }
        }
    }
    so-fpc/pic/port {
        sonet-options {
            mpls {
                pop-all-labels {
                    required-depth 2;
                }
            }
        }
    }
}
```

Using an M, MX or T Series Router Flow Collector Interface to Process and Export Multiple Flow Records

Basic passive monitoring can sometimes create a large number of flow records. However, you can manage multiple flow records with a flow collector interface. You can create a flow collector interface from a Monitoring Services II PIC. The flow collector interface combines multiple flow records received from a monitoring services interface into a compressed ASCII data file and exports the file to an FTP server.

To convert a Monitoring Services II PIC into a flow collector interface, include the **flow-collector** statement at the **[edit chassis fpc *fpc-slot* pic *pic-slot* monitoring-services application]** hierarchy level. To restore the monitoring functions of a Monitoring Services II PIC, include the **monitor** statement at the **[edit chassis fpc *fpc-slot* pic *pic-slot* monitoring-services application]** hierarchy level.

After you commit the configuration to convert the PIC between the **monitor** and **flow-collector** service types, you must take the PIC offline and then bring the PIC back online. Rebooting the router does not enable the new service type. You can use the Monitoring Services II PIC for either flow collection or monitoring, but not both types of service simultaneously.

A flow collector interface, designated by the **cp-*fpc/pic/port*** interface name, requires three logical interfaces for correct operation. Units 0 and 1 are used respectively as export channels 0 and 1 to send the compressed ASCII data files to an FTP server. You must include a class-of-service (CoS) configuration for these two

export channels to provide adequate bandwidth for file transmission. Unit 2 is used as a flow receive channel to receive flow records from a monitoring services interface.

> **NOTE:** Unlike conventional interfaces, IP addresses for flow collector logical interfaces set up a point-to-point connection between the Routing Engine and the flow collector. The **address** statement at the **[edit interfaces cp-*fpc/pic/port* unit *unit-number* family inet]** hierarchy level corresponds to the IP address of the Routing Engine. Likewise, the **destination** statement at the **[edit interfaces cp-*fpc/pic/port* unit *unit-number* family inet address *ip-address*]** hierarchy level corresponds to the IP address of the flow collector interface. As a result, you must configure the **destination** statement for Units 0 and 1 (export channels 0 and 1) with *local* addresses that can reach the FTP server. Similarly, configure the **destination** statement for Unit 2 (flow receive channel) with a *local* IP address so it can reach the monitoring services interface that sends flow records.

To activate flow collector services after the Monitoring Services II PIC is converted into a flow collector, include the **flow-collector** statement at the **[edit services]** hierarchy level. You also need to configure several additional components:

- Destination of the FTP server—Determines where the compressed ASCII data files are sent after the flow records are collected and processed. To specify the destination FTP server, include the **destinations** statement at the **[edit services flow-collector]** hierarchy level. You can specify up to two FTP server destinations and include the password for each configured server. If two FTP servers are configured, the first server in the configuration is the primary server and the second is a backup server.
- File specifications—Preset data file formats, name formats, and transfer characteristics. Files are sent by FTP to the destination FTP server when the timer expires or when a preset number of records are received, whichever comes first. To set the data file format, include the **data-format** statement at the **[edit services flow-collector file-specification *file-name*]** hierarchy level. The default data format is **flow-compressed**. To set the export timer and file size thresholds, include the **transfer** statement at the **[edit services flow-collector file-specification *file-name*]** hierarchy level and specify values for the **timeout** and **record-level** options. The default values are 600 seconds for **timeout** and 500,000 records for **record-level**.

  To set the filename format, include the **name-format** statement at the **[edit services flow-collector file-specification *file-name*]** hierarchy level. Common name format macros that you can use in your configuration are included in Table 34 on page 248.

Table 34: Name Format Macros

Field	Expansion
{*am_pm*}	**AM** or **PM**

Table 34: Name Format Macros *(continued)*

Field	Expansion
{*date*}	Expands to the current date, using the **{month}**, **{day}, and {year}** macros.
{*day*}	**01** to **31**
{*day_abbr*}	**Sun** through **Sat**
{*day_full*}	**Sunday** through **Saturday**
{*generation_number*}	Expands to a unique, sequential number for each new file created.
{*hour_12*}	**01** to **12**
{*hour_24*}	**00** to **23**
{*ifalias*}	Expands to a description string for the logical interface.
{*minute*}	**00** to **59**
{*month*}	**01** to **12**
{*month_abbr*}	**Jan** through **Dec**
{*month_full*}	**January** through **December**
{*num_zone*}	**-2359** to **+2359**
{*second*}	**00** to **60**
{*time*}	Expands to the time the file is created, using the **{hour_24}**, **{minute}**, and **{second}** macros.
{*time_zone*}	Time zone code name of the locale (**gmt, pst**, and so on).
{*year*}	**1970, 2008**, and so on.
{*year_abbr*}	**00** to **99**

- Input interface-to-flow collector interface mappings—Match an input interface with a flow collector interface and apply the preset file specifications to the input interface. To configure the default flow collector and file specifications for all input interfaces, include the **file-specification** and **collector**

statements at the **[edit services flow-collector interface-map]** hierarchy level. To override the default settings and apply flow collector and file specifications to a specific input interface, include the **file-specification** and **collector** statements at the **[edit services flow-collector interface-map *interface-name*]** hierarchy level.

- Transfer log settings—Allow you to configure the destination FTP server where log files containing the transfer activity history for a flow collector interface are to be archived, the name for the log file, and the amount of time the router waits before sending the log file to the FTP server. To configure, include the **archive-sites**, **filename-prefix**, and **maximum-age** statements at the **[edit services flow-collector transfer-log-archive]** hierarchy level. The default value for the **maximum-age** statement is 120 minutes, with a range of 1 to 360 minutes. Also, you can configure up to five FTP archive site servers to receive log files.
- Miscellaneous settings—Allow you to configure values for the IP address of the analyzer, an identifier for the analyzer, the maximum number of times the flow collector interface attempts to send transfer log files to the FTP server, and the amount of time the flow collector interface waits between retry attempts. To configure, include the **analyzer-address**, **analyzer-id**, **retry**, and **retry-delay** statements at the **[edit services flow-collector]** hierarchy level. The range for the **retry** statement is 0 through 10 retry attempts. The default for the **retry-delay** statement is 30 seconds and the range is 0 through 60 seconds.

To specify a flow collector interface as the destination for flow records coming from a Monitoring Services or Monitoring Services II PIC, include the **collector-pic** statement at the **[edit forwarding-options monitoring *group-name* family inet output flow-export-destination]** hierarchy level. You can select either the flow collector interface or a flow server as the destination for flow records, but you cannot select both destination types simultaneously.

There is also a Juniper Networks enterprise Management Information Base (MIB) for the flow collector interface. The Flow Collector Services MIB allows you to use SNMP to monitor the flow collector interface. The MIB provides statistics on files, records, memory, FTP, and error states of a flow collector interface. It also provides SNMP traps for unavailable destinations, unsuccessful file transfers, flow overloading, and memory overloading. For more information, see the *Junos Network Management Configuration Guide* or view the enterprise-specific Juniper Networks MIBs at **https://www.juniper.net/techpubs/software/junos/mibs.html**.

In summary, to implement the flow collector service, include statements at the **[edit chassis]**, **[edit interfaces]**, **[edit forwarding-options]**, and **[edit services]** hierarchy levels. The excerpt on the following pages shows the flow collector service configuration hierarchy. For a full configuration example, see "Example: Configuring a Flow Collector Interface on an M, MX or T Series Router" on page 253.

```
[edit]
chassis {
    fpc fpc-slot {
        pic pic-slot {
            monitoring-services {
                application flow-collector;
```

```
        }
      }
    }
  }
  interfaces {
    cp-fpc/pic/port {
      description ”flow_collector_interface”;
      unit 0 {
        family inet {
          address ip-address {
            destination  ip-address;
          }
        }
      }
      unit 1 {
        family inet {
          address ip-address {
            destination ip-address;
          }
        }
      }
      unit 2 {
        family inet {
          address ip-address {
            destination ip-address;
          }
        }
      }
    }
    interface-fpc/pic/port {
      description “export_interface”;
      unit 0 {
        family inet {
          address ip-address;
        }
      }
    }
    mo-fpc/pic/port  {
      description “monitoring_services_interface”;
      unit 0 {
        family inet;
      }
    }
    SONET/SDH, ATM2 IQ, or Ethernet-based-interface-fpc/pic/port {
```

```
        description " input_interface";
        encapsulation encapsulation-type;
        passive-monitor-mode; # Apply to the logical interface for SONET/SDH
    }
}
forwarding-options {
    monitoring group1 {
        family inet {
            output {
                export-format cflowd-version-5;
                flow-active-timeout value;
                flow-inactive-timeout value;
                flow-export-destination collector-pic;
                interface mo-fpc/pic/port {
                    source-address ip-address;
                }
            }
        }
    }
}
services {
    flow-collector {
        analyzer-address ip-address;
        analyzer-id name;
        retry value;
        retry-delay seconds;
        destinations {
            "ftp://username@ftp-server-address-1//directory/" {
                password "encrypted-password";
            }
            "ftp://username@ftp-server-address-2//directory/" {
                password "encrypted-password";
            }
        }
        file-specification {
            file-specification-name {
            }
            data-format flow-compressed;
            transfer timeout value record-level size;
        }
    }
    interface-map {
        file-specification file-specification-name;
        collector cp-fpc/pic/port;
```

```
        interface-name {
            file-specification file-specification-name;
            collector cp-fpc/pic/port;
        }
    }
    transfer-log-archive {
        filename-prefix filename;
        maximum-age timeout-value;
        archive-sites {
            "ftp://username@ip-address//directory/" {
                password "encrypted-password";
            }
        }
    }
}
```

Example: Configuring a Flow Collector Interface on an M, MX or T Series Router

Figure 28: Flow Collector Interface Topology Diagram

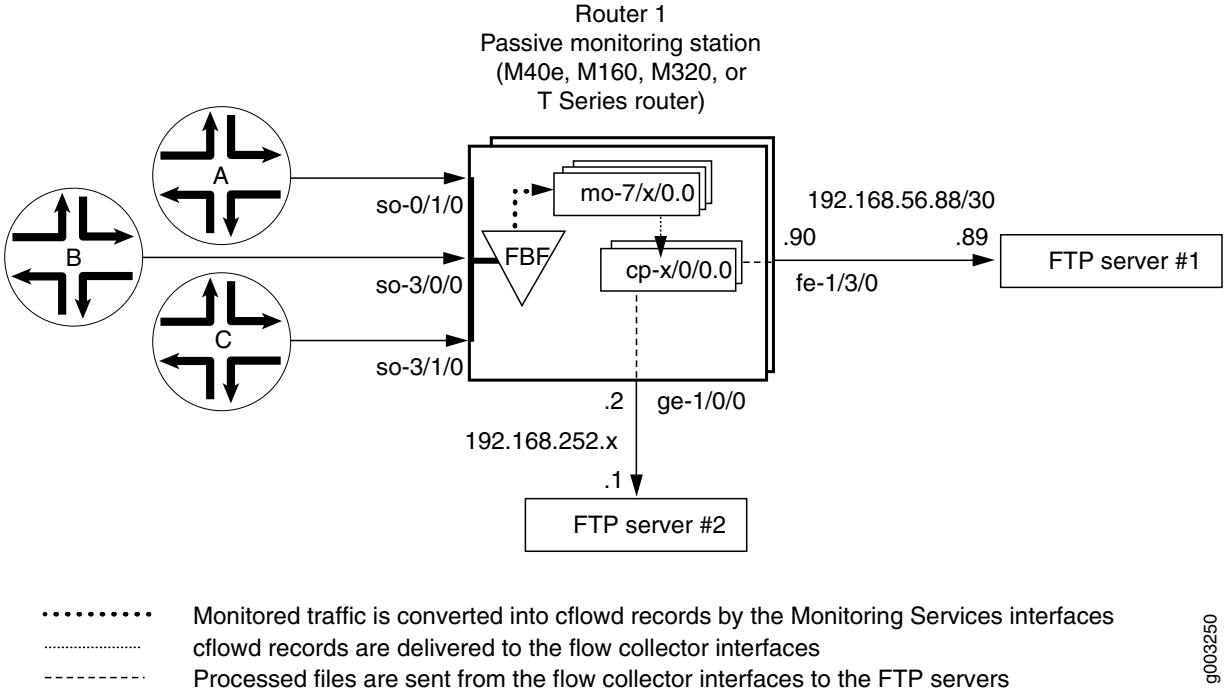


Figure 28 on page 253 shows the path traveled by monitored traffic as it passes through the router. Packets arrive at input interfaces **so-0/1/0**, **so-3/0/0**, and **so-3/1/0**. The raw packets are directed into a filter-based forwarding routing instance and processed into flow records by the monitoring services interfaces **mo-7/1/0**, **mo-7/2/0**, and **mo-7/3/0**. The flow records are compressed into files at the flow collector interfaces **cp-6/0/0** and **cp-7/0/0** and sent to the FTP server for analysis. Finally, a mandatory class-of-service (CoS) configuration is applied to export channels 0 and 1 on the flow collector interfaces to manage the outgoing processed files.

Router 1

```
[edit]
chassis {
    fpc 6 {
        pic 0 {
            monitoring-services {
                application flow-collector; # This converts a Monitoring Services II PIC
            }       # into a flow collector interface.
        }
    }
    fpc 7 {
        pic 0 {
            monitoring-services {
                application flow-collector; # This converts a Monitoring Services II PIC
            }         # into a flow collector interface.
        }
    }
}
interfaces {
    cp-6/0/0 {
        unit 0 {# Logical interface .0 on a flow collector interface is export
            family inet { # channel 0 and sends records to the FTP server.
                filter {
                    output cp-ftp; # Apply the CoS filter here.
                }
                address 10.0.0.1/32 {
                    destination 10.0.0.2;
                }
            }
        }
        unit 1 { # Logical interface .1 on a flow collector interface is export
            family inet { # channel 1 and sends records to the FTP server.
                filter {
```

```
            output cp-ftp; # Apply the CoS filter here.
          }
          address 10.1.1.1/32 {
            destination 10.1.1.2;
          }
        }
      }
      unit 2 { # Logical interface .2 on a flow collector interface is the flow
        family inet { # receive channel that communicates with the Routing Engine.
          address 10.2.2.1/32 { # Do not apply a CoS filter on logical interface .2.
            destination 10.2.2.2;
          }
        }
      }
    }
    cp-7/0/0 {
      unit 0 { # Logical interface .0 on a flow collector interface is export
        family inet { # channel 0 and sends records to the FTP server.
          filter {
            output cp-ftp; # Apply the CoS filter here.
          }
          address 10.3.3.1/32 {
            destination 10.3.3.2;
          }
        }
      }
      unit 1 { # Logical interface .1 on a flow collector interface is export
        family inet { # channel 1 and sends records to the FTP server.
          filter {
            output cp-ftp; # Apply the CoS filter here.
          }
          address 10.4.4.1/32 {
            destination 10.4.4.2;
          }
        }
      }
      unit 2 { # Logical interface .2 on a flow collector interface is the flow
        family inet { # receive channel that communicates with the Routing Engine.
          address 10.5.5.1/32 { # Do not apply a CoS filter on logical interface .2.
            destination 10.5.5.2;
          }
        }
```

```
    }
}
fe-1/3/0 { # This is the exit interface leading to the first FTP server.
    unit 0 {
        family inet {
            address 192.168.56.90/30;
        }
    }
}
ge-1/0/0 { # This is the exit interface leading to the second FTP server.
    unit 0 {
        family inet {
            address 192.168.252.2/24;
        }
    }
}
mo-7/1/0 {  # This is the first interface that creates flow records.
    unit 0 {
        family inet;
    }
}
mo-7/2/0 { # This is the second interface that creates flow records.
    unit 0 {
        family inet;
    }
}
mo-7/3/0 { # This is the third interface that creates flow records.
    unit 0 {
        family inet;
    }
}
so-0/1/0 { # This is the first input interface that receives traffic to be monitored.
    encapsulation ppp;
    unit 0 {
        passive-monitor-mode; # This allows the interface to be passively monitored.
        family inet {
            filter {
                input catch; # The filter-based forwarding filter is applied here.
            }
        }
    }
}
```

```
    so-3/0/0 { # This is the second interface that receives traffic to be monitored.
        encapsulation ppp;
        unit 0 {
            passive-monitor-mode; # This allows the interface to be passively monitored.
            family inet {
                filter {
                    input catch; # The filter-based forwarding filter is applied here.
                }
            }
        }
    }
    so-3/1/0 { # This is the third interface that receives traffic to be monitored.
        encapsulation ppp;
        unit 0 {
            passive-monitor-mode; # This allows the interface to be passively monitored.
            family inet {
                filter {
                    input catch; # The filter-based forwarding filter is applied here.
                }
            }
        }
    }
}
forwarding-options {
    monitoring group1 { # Always define your monitoring group here.
        family inet {
            output {
                export-format cflowd-version-5;
                flow-active-timeout 60;
                flow-inactive-timeout 15;
                flow-export-destination collector-pic; # Sends records to the flow collector.
                interface mo-7/1/0.0 {
                    source-address 192.168.252.2;
                }
                interface mo-7/2/0.0 {
                    source-address 192.168.252.2;
                }
                interface mo-7/3/0.0 {
                    source-address 192.168.252.2;
                }
            }
        }
```

```
    }
}
routing-options {
    interface-routes {
        rib-group inet common;
    }
    rib-groups {
        common {
            import-rib [ inet.0 fbf_instance.inet.0 ];
        }
    }
    forwarding-table {
        export pplb;
    }
}
policy-options {
    policy-statement pplb {
        then {
            load-balance per-packet;
        }
    }
}
class-of-service { # A class-of-service configuration for the flow collector interface
    interfaces { # is mandatory when implementing flow collector services.
        cp-6/0/0 {
            scheduler-map cp-map;
        }
        cp-7/0/0 {
            scheduler-map cp-map;
        }
    }
    scheduler-maps {
        cp-map {
            forwarding-class best-effort scheduler Q0;
            forwarding-class expedited-forwarding scheduler Q1;
            forwarding-class network-control scheduler Q3;
        }
    }
    schedulers {
        Q0 {
            transmit-rate remainder;
            buffer-size percent 90;
```

```
        }
        Q1 {
            transmit-rate percent 5;
            buffer-size percent 5;
            priority strict-high;
        }
        Q3 {
            transmit-rate percent 5;
            buffer-size percent 5;
        }
    }
}
firewall {
    family inet {
        filter cp-ftp { # This filter provides CoS for flow collector interface traffic.
            term t1 {
                then forwarding-class expedited-forwarding;
            }
        }
    }
    filter catch { # This firewall filter sends incoming traffic into the
        interface-specific; # filter-based forwarding routing instance.
        term def {
            then {
                count counter;
                routing-instance fbf_instance;
            }
        }
    }
}
routing-instances {
    fbf_instance { # This instance sends traffic to the monitoring services interface.
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop mo-7/1/0.0;
            }
        }
    }
}
services {
    flow-collector { # Define properties for flow collector interfaces here.
```

```
    analyzer-address 10.10.10.1; # This is the IP address of the analyzer.
    analyzer-id server1; # This helps to identify the analyzer.
    retry 3; # Maximum number of attempts by the PIC to send a file transfer log.
    retry-delay 30; # The time interval between attempts to send a file transfer log.
    destinations { # This defines the FTP servers that receive flow collector output.
      "ftp://user@192.168.56.89//tmp/collect1/" {  # The primary FTP server.
        password "$ABC123"; # SECRET-DATA
      }
      "ftp://user@192.168.252.1//tmp/collect2/" { # The second FTP server.
        password "$ABC123"; # SECRET-DATA
      }
    }
    file-specification {  # Define sets of flow collector characteristics here.
      def-spec {
      }
      data-format flow-compressed; # The default compressed output format.
    }
    f1 {
      name-format "cFlowd-py69Ni69-0-%D_%T-%I_%N.bcp.bi.gz";
      data-format flow-compressed; # The default compressed output format.
      transfer timeout 1800 record-level 1000000; # Here are configured values.
    }
}
interface-map { # Allows you to map interfaces to flow collector interfaces.
    file-specification def-spec; # Flows generated for default traffic are sent to the
    collector cp-7/0/0; # default flow collector interface cp-7/0/0.
    so-0/1/0.0 {# Flows generated for the so-0/1/0 interface are sent
      collector cp-6/0/0; # to cp-6/0/0, and the file-specification used is “default”.
    }
    so-3/0/0.0 { # Flows generated for the so-3/0/0 interface are sent
      file-specification f1; # to cp-6/0/0, and the file-specification used is "f1."
      collector cp-6/0/0;
    }
    so-3/1/0.0; # Because no settings are defined, flows generated for this
}
transfer-log-archive { # Sends flow collector interface log files to an FTP server.
    filename-prefix so_3_0_0_log;
    maximum-age 15;
    archive-sites {
      "ftp://user@192.168.56.89//tmp/transfers/" {
        password "$ABC123";
      }
```

```
        }
      }
    }
```

Verifying Your Work

To verify that your flow collector configuration is working, use the following commands on the monitoring station that is configured for flow collection:

- **clear services flow-collector statistics**
- **request services flow-collector change-destination (primary | secondary)**
- **request services flow-collector test-file-transfer**
- **show services flow-collector file interface (detail | extensive | terse)**
- **show services flow-collector (detail | extensive)**
- **show services flow-collector input interface (detail | extensive | terse)**

The following section shows the output of the **show** commands used with the configuration example:

`user@router1>` **show services flow-collector input interface cp-6/0/0 detail**

```
Interface                      Packets        Bytes
mo-7/1/0.0                        6170      8941592

user@router1>   show services flow-collector interface all detail
Flow collector interface: cp-6/0/0
Interface state: Collecting flows
  Packets     Bytes     Flows Uncompressed  Compressed     FTP bytes FTP files
                                    Bytes       Bytes
     6736   9757936    195993     21855798     3194148             0         0
Flow collector interface: cp-7/0/0
Interface state: Collecting flows
  Packets     Bytes     Flows Uncompressed  Compressed     FTP bytes FTP files
                                    Bytes       Bytes
        0         0         0            0           0             0         0

user@router1>   show services flow-collector input interface cp-6/0/0 extensive
Interface                      Packets        Bytes
mo-7/1/0.0                        6260      9074096
```

```
user@router1>   show services flow-collector interface cp-6/0/0 extensive
Flow collector interface: cp-6/0/0
Interface state: Collecting flows
Memory:
    Used: 19593212, Free: 479528656
Input:
    Packets: 6658, per second: 0, peak per second: 0
    Bytes: 9647752, per second: 12655, peak per second: 14311
    Flow records processed: 193782, per second: 252, peak per second: 287
Allocation:
    Blocks allocated: 174, per second: 0, peak per second: 0
    Blocks freed: 0, per second: 0, peak per second: 0
    Blocks unavailable: 0, per second: 0, peak per second: 0
Files:
    Files created: 1, per second: 0, peak per second: 0
    Files exported: 0, per second: 0, peak per second: 0
    Files destroyed: 0, per second: 0, peak per second: 0
Throughput:
    Uncompressed bytes: 21075152, per second: 52032, peak per second: 156172
    Compressed bytes: 3079713, per second: 7618, peak per second: 22999
Packet drops:
    No memory: 0, Not IP: 0
    Not IPv4: 0, Too small: 0
    Fragments: 0, ICMP: 0
    TCP: 0, Unknown: 0
    Not JUNOS flow: 0
File Transfer:
    FTP bytes: 0, per second: 0, peak per second: 0
    FTP files: 0, per second: 0, peak per second: 0
    FTP failure: 0
Export channel: 0
    Current server: Secondary
    Primary server state: OK, Secondary server state: OK
Export channel: 1
    Current server: Secondary
    Primary server state: OK, Secondary server state: OK

user@router1>   show services flow-collector file interface cp-6/0/0 terse
File name                                                  Flows State
cFlowd-py69Ni69-0-20031112_014301-so_3_0_0_0.bcp.bi.gz          185643 Active

user@router1>   show services flow-collector file interface cp-6/0/0 detail
Filename: cFlowd-py69Ni69-0-20031112_014301-so_3_0_0_0.bcp.bi.gz
```

```
  Throughput:
    Flow records: 187067, Uncompressed bytes: 21121960, Compressed bytes: 2965643

  Status:
    State: Active, Transfer attempts: 0

user@router1>  show services flow-collector file interface cp-6/0/0 extensive
Filename: cFlowd-py69Ni69-0-20031112_014301-so_3_0_0_0.bcp.bi.gz
  Throughput:
    Flow records: 188365, per second: 238, peak per second: 287
    Uncompressed bytes: 21267756, per second: 27007, peak per second: 32526
    Compressed bytes: 2965643, per second: 0, peak per second: 22999
  Status:
    Compressed blocks: 156, Block count: 156
    State: Active, Transfer attempts: 0
```

To clear statistics for a flow collector interface, issue the **clear services flow-collector statistics interface (all | *interface-name*)** command.

Another useful flow collector option allows you to change the FTP server from primary to secondary and test for FTP transfers. To force the flow collector interface to use a primary or secondary FTP server, include the **primary** or **secondary** option when you issue the **request services flow-collector change-destination interface cp-*fpc*/*pic*/*port*** command.

If you configure only one primary server and issue this command with the **primary** option, you receive the error message "Destination change not needed." If the secondary server is not configured and you issue this command with the **secondary** option, you receive the error message "Destination not configured." Otherwise, when both servers are configured properly, successful output appears as follows.

```
user@router1> request services flow-collector change-destination interface  cp-6/0/0 primary

Flow collector interface: cp-6/0/0
Interface state: Collecting flows
Destination change successful

user@router1>  request services flow-collector change-destination interface  cp-6/0/0 secondary
Flow collector interface: cp-6/0/0
Interface state: Collecting flows
Destination change successful
```

Other options for the **request services flow-collector change-destination interface cp-*fpc*/*pic*/*port*** command are **immediately** (which forces an instant switchover), **gracefully** (the default behavior that allows

a gradual switchover), **clear-files** (which purges existing data files), and **clear-logs** (which purges existing log files).

To verify that transfer log files are being scheduled for delivery to the FTP servers, issue the **request services flow-collector test-file-transfer *filename* interface cp-*fpc*/*pic*/*port*** command. Include the desired export channel (zero or one) and target FTP server (primary or secondary) with this command.

```
user@router1> request services flow-collector test-file-transfer test_file  interface cp-6/0/0
channel-one primary
```

```
Flow collector interface: cp-6/0/0
Interface state: Collecting flows
Response: Test file transfer successfully scheduled
```

Another way you can check for the success of your file transfers is by analyzing the transfer log. A transfer log sends detailed information about files that are collected and processed by the flow collector interface. Table 35 on page 264 explains the various fields available in the transfer log.

Table 35: Flow Collector Interface Transfer Log Fields

Field	Explanation
fn	Filename
sz	File size
nr	Number of records
ts	Timestamp with the format of year (4 digits), month (2 digits), day (2 digits), hours (2 digits), minutes (2 digits), and seconds (2 digits).
sf	Success flag—The values are **1** for success and **0** for failure.
ul	Server URL
rc	FTP result code
er	FTP error text
tt	Transfer time

This is an example of a successful transfer log:

```
fn="cFlowd-py69Ni69-0-20040227_230438-at_4_0_0_4_3.bcp.bi.gz":sz=552569
:nr=20000:ts="20040227230855":sf=1:ul="ftp://10.63.152.1/tmp/server1/:"rc=250:
er="":tt=3280
```

This is an example of a transfer log when an FTP session fails:

```
fn="cFlowd-py69Ni69-0-20040227_230515-at_4_0_0_2_8.bcp.bi.gz":sz=560436
:nr=20000:ts="20040227230855":sf=1:ul="ftp://10.63.152.1/tmp/server1/:"rc=250
:er="":tt=3290
```

As the flow collector interface receives and processes flow records, the PIC services logging process (fsad) handles the following tasks:

- When the flow collector interface transfers a file to the FTP server, a temporary log file is created in the **/var/log/flowc** directory. The temporary log file has this filenaming convention:

  <***hostname***>_<***filename_prefix***>_ ***YYYYMMDD_hhmmss*.tmp**

  hostname is the hostname of the transfer server, ***filename_prefix*** is the same value defined with the **filename-prefix** statement at the [**edit services flow-collector transfer-log-archive**] hierarchy level, ***YYYYMMDD*** is the year, month, and date, and ***hhmmss*** is the timestamp indicating hours, minutes, and seconds.

- After the log file has been stored in the router for the length of time specified by the **maximum-age** statement at the [**edit services flow-collector transfer-log-archive**] hierarchy level (the default is 120 minutes), the temporary log file is converted to an actual log file and the temporary file is deleted. The new log file retains the same naming conventions, except the extension is **.log*.

- When the final log file is created and compressed, the PIC services logging process (fsad) tries to send the log file from the **/var/log/flowc** directory to an FTP server. You can specify up to five FTP servers to receive the log files by including the **archive-sites** statement at the [**edit services flow-collector transfer-log-archive**] hierarchy level. The logging process attempts to send the log file to one server at a time, in order of their appearance in the configuration. Upon the first successful transfer, the log file is deleted and the logging process stops sending log files to the remaining FTP servers in the list.

- If the log file transfer is not successful, the log file is moved to the **/var/log/flowc/failed** directory. Every 30 minutes, the logging process tries to resend the log files. After the log files are transferred successfully, they are deleted from the **/var/log/flowc/failed** directory.

> **NOTE:** If the memory for a flow collector interface is full, the interface might drop incoming packets.

After the flow collector interface successfully delivers the processed information file to the FTP server, you can analyze the file. The file contains detailed information about the flows collected and processed by the flow collector interface. Table 36 on page 266 explains the various fields available in the flow collector interface file.

Table 36: Flow Collector Interface File Fields in Order of Appearance

Field	Explanation
linkDir	Link directory—A randomly generated number used to identify the record
analyzer-address	Analyzer address
analyzer-ID	Analyzer identifier
ifAlias	Interface identifier
source-address	Source address
destination-address	Destination address
packets	Number of packets
bytes	Number of bytes
start-time	Start time
end-time	End time
source-port	Source port
destination-port	Destination port
tcp_flag	TCP flag
protocol	IP protocol number
src_AS_number	Source AS number

Table 36: Flow Collector Interface File Fields in Order of Appearance *(continued)*

Field	Explanation
dst_AS_number	Destination AS number

This is an example of output from a flow collector interface file:

```
1179924161237455778 2|10.10.10.1|server1|at_4_0_0_4|192.168.10.100|10.0.0.1|8|
3136|1077926402|1077926402|8224|12336|27|6|0|0
```

Copying and Redirecting Traffic with Port Mirroring and Filter-Based Forwarding

This section discusses additional techniques you can use with the passive flow monitoring application:

- In addition to flow analysis, you can analyze a copy of the original traffic with a single packet analyzer. To implement this technique, divert traffic with a filter-based forwarding routing instance and send the monitored traffic through a physical interface to the packet analyzer.
- You can cluster the traffic into different groups and redirect this traffic to multiple packet analyzers. For example, you can break traffic flows into TCP groups and UDP groups and send these groups of packets to different analyzers. To accomplish this, you use port mirroring and send a copy of the original traffic to a Tunnel PIC. Then you can apply a firewall filter, split the traffic into your desired groups, and send these groups toward different exit interfaces leading to the packet analyzers. This technique provides maximum flexibility for traffic analysis.
- For secure transmission of the copied or grouped traffic, you can encrypt the diverted traffic with an ES PIC and send this traffic to a packet analyzer over an IP Security (IPSec) tunnel.

To implement the filter-based forwarding enhancement methods, see the following sections:

- Specifying Port Mirroring Input and Output on M, MX or T Series Routers on page 230
- Creating a Firewall Filter on an M, MX or T Series Router to Split the Port-Mirrored Traffic into Different Instances on page 232
- Applying the Firewall Filter to a Tunnel PIC Interface on page 233
- Using Filter-Based Forwarding to Export Monitored Traffic to Multiple Destinations on page 233
- Configuring a Routing Table Group on an M, MX or T Series Router to Add Interface Routes into the Forwarding Instance on page 234

- Using IPSec and an ES PIC on an M, MX or T Series Router to Send Encrypted Traffic to a Packet Analyzer on page 235
- Applying a Firewall Filter Output Interface on an M, MX or T Series Router to Port-mirror Traffic to PICs or Flow Collection Services on page 237

CHAPTER 4

Processing and Exporting Multiple Records Using Flow Collection

IN THIS CHAPTER

Flow Collection Overview

You can process and export multiple cflowd records with a flow collector interface. You create a flow collector interface on a Monitoring Services II or Multiservices 400 PIC. The flow collector interface combines multiple cflowd records into a compressed ASCII data file and exports the file to an FTP server. To convert a services PIC into a flow collector interface, include the **flow-collector** statement at the **[edit chassis fpc *fpc-slot* pic *pic-slot* monitoring-services application]** hierarchy level.

You can use the services PIC for either flow collection or monitoring, but not for both types of service simultaneously. When converting the PIC between service types, you must configure the **flow-collector** statement, take the PIC offline, and then bring the PIC back online. Restarting the router does not enable the new service type.

A flow collector interface, designated by the **cp-*fpc*/*pic*/*port*** interface name, requires three logical interfaces for correct operation. Units 0 and 1 are used to send the compressed ASCII data files to an FTP server, while Unit 2 is used to receive cflowd records from a monitoring services interface.

> **NOTE:** Unlike conventional interfaces, the **address** statement at the **[edit interfaces cp-*fpc/pic/port* unit *unit-number* family inet]** hierarchy level corresponds to the IP address of the Routing Engine. Likewise, the **destination** statement at the **[edit interfaces cp-*fpc/pic/port* unit *unit-number* family inet address *ip-address*]** hierarchy level corresponds to the IP address of the flow collector interface. As a result, you must configure the **destination** statement for Unit 0 and 1 with *local* addresses that can reach the FTP server. Similarly, configure the **destination** statement for Unit 2 with a *local* IP address so it can reach the monitoring services interface that sends cflowd records.

To activate flow collector services after the services PIC is converted into a flow collector, include the **flow-collector** statement at the **[edit services]** hierarchy level.

After you activate the flow collector, you need to configure the following components:

- Destination of the FTP server
- File specifications
- Input interface-to-flow collector interface mappings
- Transfer log settings

RELATED DOCUMENTATION

Configuring Flow Collection | 2701

Sending cflowd Records to Flow Collector Interfaces | 282

Configuring Flow Collection Mode and Interfaces on Router Services PICs on M and T Series Routers | 283

Configuring Flow Collection

IN THIS SECTION

This section describes the following tasks for configuring flow collection:

Configuring Destination FTP Servers for Flow Records

Flow collection destinations are where the compressed ASCII data files are sent after the cflowd records are collected and processed. To specify the destination FTP server, include the **destinations** statement at the **[edit services flow-collector]** hierarchy level. You can specify up to two FTP server destinations and include the password for each configured server. If two FTP servers are configured, the first server in the configuration is the primary server and the second is a backup server.

To configure a destination for flow collection files, include the **destinations** statement at the **[edit services flow-collector]** hierarchy level:

```
[edit services flow-collector]
destinations {
    ftp:url {
        password "password";
    }
}
```

To specify the destination FTP server, include the **ftp:*url*** statement. The value ***url*** is the FTP server address for the primary flow collection destination and can include macros.

When you include macros in the **ftp:*url*** statement, a directory can be created only for a single level. For example, the path **ftp://10.2.2.2/%m/%Y** expands to **ftp://10.2.2.2/01/2005**, and the software attempts to create the directory **01/2005** on the destination FTP server. If the **01/** directory already exists on the destination FTP server, the software creates the **/2005/ directory** one level down. If the **01/** directory does not exist on the destination FTP server, the software cannot create the **/2005/ directory**, and the FTP server destination fails. For more information about macros, see **ftp**.

To specify the FTP server password, include the **password "*password*"** statement. The password must be enclosed in quotation marks. You can specify up to two destination FTP servers. The first destination specified is considered the primary destination.

Configuring a Packet Analyzer

You can specify values for the IP address and identifier of a packet analyzer to which the flow collector interface sends traffic for analysis. The values you specify here override any default values configured elsewhere.

To configure an IP address and identifier for the packet analyzer, include the **analyzer-address** and **analyzer-id** statements at the **[edit services flow-collector]** hierarchy level:

```
[edit services flow-collector]
analyzer-address address;
analyzer-id name;
```

Configuring File Formats

You configure data file formats, name formats, and transfer characteristics for the flow collection files. File records are sent to the destination FTP server when the timer expires or when a preset number of records are received, whichever comes first.

To configure the flow collection file format, include the **file-specification** statement at the **[edit services flow-collector]** hierarchy level:

```
[edit services flow-collector]
file-specification {
   variant variant-number {
      data-format format;
      name-format format;
      transfer {
         record-level number;
         timeout seconds;
      }
   }
}
```

To set the data file format, include the **data-format** statement. To set the file name format, include the **name-format** statement. To set the export timer and file size thresholds, include the **transfer** statement and specify values for the **timeout** and **record-level** options.

For example, you can specify the name format as follows:

```
[edit services flow-collector file-specification variant variant-number]
name-format "cFlowd-py69Ni69-0-%D_%T-%I_%N.bcp.bi.gz";
```

In this example, **cFlowd-py69Ni69-0** is the static portion used verbatim, **%D** is the date in YYYYMMDD format, **%T** is the time in HHMMSS format, **%I** is the value of **ifAlias**, **%N** is the generation number, and **bcp.bi.gz** is a user-configured string. A number of macros are supported for expressing the date and time information in different ways; for a complete list, see the summary section for **name-format**.

Configuring Interface Mappings

You can match an input interface with a flow collector interface and apply the preset file specifications to the input interface.

To configure an interface mapping, include the **interface-map** statement at the **[edit services flow-collector]** hierarchy level:

```
[edit services flow-collector]
interface-map {
    collector interface-name;
    file-specification variant-number;
    interface-name {
        collector interface-name;
        file-specification variant-number;
    }
}
```

To configure the default flow collector and file specifications for all input interfaces, include the **file-specification** and **collector** statements at the **[edit services flow-collector interface-map]** hierarchy level. To override the default settings and apply flow collector and file specifications to a specific input interface, include the **file-specification** and **collector** statements at the **[edit services flow-collector interface-map *interface-name*]** hierarchy level.

Configuring Transfer Logs

You can configure the filename, export interval, maximum size, and destination FTP server for log files containing the transfer activity history for a flow collector interface.

To configure a transfer log, include the **transfer-log-archive** statement at the **[edit services flow-collector]** hierarchy level:

```
[edit services flow-collector]
transfer-log-archive {
    archive-sites {
        ftp:url {
            password "password";
            username username;
```

```
        }
    }
    filename-prefix prefix;
    maximum-age minutes;
}
```

To configure the destination for archiving files, include the **archive-sites** statement. Specify the filename as follows:

```
[edit services flow-collector transfer-log]
filename "cFlowd-py69Ni69-0-%D_%T";
```

where **cFlowd-py69Ni69-0** is the static portion used verbatim, **%D** is the date in YYYYMMDD format, and **%T** is the time in HHMMSS format.

You can optionally include the following statements:

- **filename-prefix**—Sets a standard prefix for all the logged files.
- **maximum-age**—Specifies the duration a file remains on the server. The range is 1 through 360 minutes.

Configuring Retry Attempts

You can specify values for situations in which the flow collector interface needs more than one attempt to transfer log files to the FTP server:

- Maximum number of retry attempts
- Amount of time the flow collector interface waits between successive retries

To configure retry settings, include the **retry** and **retry-delay** statements at the **[edit services flow-collector]** hierarchy level:

```
retry number;
retry-delay seconds;
```

The **retry** value can be from 0 through 10. The **retry-delay** value can be from 0 through 60 seconds.

RELATED DOCUMENTATION

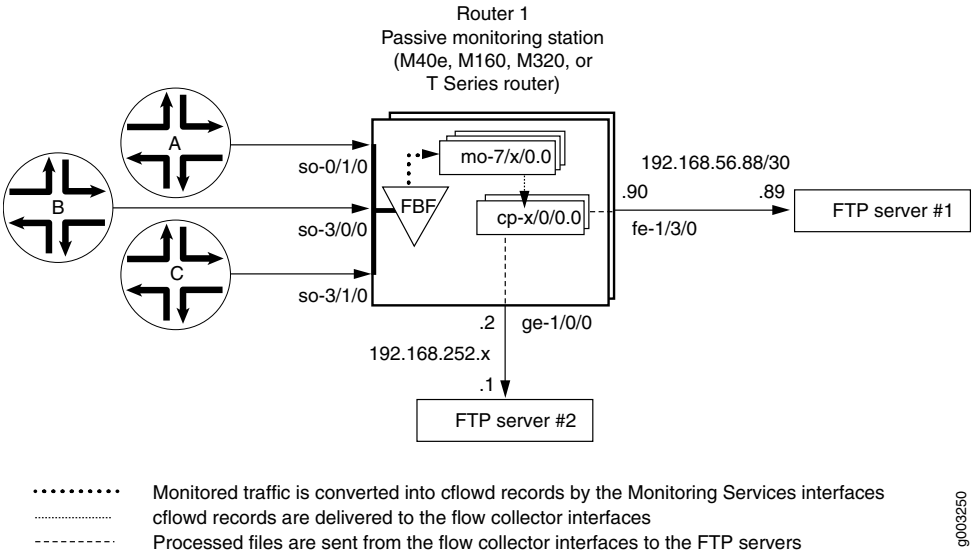
Flow Collection Overview | 269

Sending cflowd Records to Flow Collector Interfaces | 282

Example: Configuring Flow Collection

Figure 29 on page 275 shows the path traveled by monitored traffic as it passes through the router. Packets arrive at input interfaces so-0/1/0, so-3/0/0, and so-3/1/0. The raw packets are directed into a filter-based forwarding routing instance and processed into cflowd records by the monitoring services interfaces mo-7/1/0, mo-7/2/0, and mo-7/3/0. The cflowd records are compressed into files at the flow collector interfaces cp-6/0/0 and cp-7/0/0 and sent to the FTP server for analysis. Finally, a mandatory class-of-service (CoS) configuration is applied to export channels 0 and 1 on the flow collector interfaces to manage the outgoing processed files.

Figure 29: Flow Collector Interface Topology Diagram




```
[edit]
chassis {
   fpc 6 {
      pic 0 {
         monitoring-services {
            application flow-collector; # This converts a Monitoring Services II or
                        # Multiservices 400 PIC into a flow collector interface.
         }
      }
```

```
    }
    fpc 7 {
        pic 0 {
            monitoring-services {
                application flow-collector; # This converts a Monitoring Services II or
                                # Multiservices 400 PIC into a flow collector interface.
            }
        }
    }
}
interfaces {
    cp-6/0/0 {
        unit 0 { # Logical interface .0 on a flow collector interface is export
            family inet { # channel 0 and sends records to the FTP server.
                filter {
                    output cp-ftp; # Apply the CoS filter here.
                }
                address 10.0.0.1/32 {
                    destination 10.0.0.2;
                }
            }
        }
        unit 1 { # Logical interface .1 on a flow collector interface is export
            family inet {# channel 1 and sends records to the FTP server.
                filter {
                    output cp-ftp; # Apply the CoS filter here.
                }
                address 10.1.1.1/32 {
                    destination 10.1.1.2;
                }
            }
        }
        unit 2 { # Logical interface .2 on a flow collector interface is the flow
            family inet { # receive channel that communicates with the Routing Engine.
                address 10.2.2.1/32 { # Do not apply a CoS filter on logical interface .2.
                    destination 10.2.2.2;
                }
            }
        }
    }
    cp-7/0/0 {
        unit 0 {# Logical interface .0 on a flow collector interface is export
            family inet {# channel 0 and sends records to the FTP server.
                filter {
```

```
            output cp-ftp;# Apply the CoS filter here.
          }
          address 10.3.3.1/32 {
            destination 10.3.3.2;
          }
        }
      }
      unit 1 {# Logical interface .1 on a flow collector interface is export
        family inet {# channel 1 and sends records to the FTP server.
          filter {
            output cp-ftp;# Apply the CoS filter here.
          }
          address 10.4.4.1/32 {
            destination 10.4.4.2;
          }
        }
      }
      unit 2 {# Logical interface .2 on a flow collector interface is the flow
        family inet {# receive channel that communicates with the Routing Engine.
          address 10.5.5.1/32 {# Do not apply a CoS filter on logical interface .2.
            destination 10.5.5.2;
          }
        }
      }
}
fe-1/3/0 { # This is the exit interface leading to the first FTP server.
    unit 0 {
      family inet {
        address 192.168.56.90/30;
      }
    }
}
ge-1/0/0 { # This is the exit interface leading to the second FTP server.
    unit 0 {
      family inet {
        address 192.168.252.2/24;
      }
    }
}
mo-7/1/0 { # This is the first interface that creates cflowd records.
    unit 0 {
      family inet;
    }
}
```

```
    mo-7/2/0 { # This is the second interface that creates cflowd records.
        unit 0 {
            family inet;
        }
    }
    mo-7/3/0 { # This is the third interface that creates cflowd records.
        unit 0 {
            family inet;
        }
    }
    so-0/1/0 { # This is the first input interface that receives traffic to be monitored.
        encapsulation ppp;
        unit 0 {
            passive-monitor-mode; # This allows the interface to be passively monitored.
            family inet {
                filter {
                    input catch; # The filter-based forwarding filter is applied here.
                }
            }
        }
    }
    so-3/0/0 { # This is the second interface that receives traffic to be monitored.
        encapsulation ppp;
        unit 0 {
            passive-monitor-mode; # This allows the interface to be passively monitored.
            family inet {
                filter {
                    input catch; # The filter-based forwarding filter is applied here.
                }
            }
        }
    }
so-3/1/0 { # This is the third interface that receives traffic to be monitored.
    encapsulation ppp;
    unit 0 {
        passive-monitor-mode; # This allows the interface to be passively monitored.
        family inet {
            filter {
                input catch; # The filter-based forwarding filter is applied here.
            }
        }
    }
}
forwarding-options {
```

```
    monitoring group1 {# Always define your monitoring group here.
        family inet {
            output {
                export-format cflowd-version-5;
                flow-active-timeout 60;
                flow-inactive-timeout 15;
                flow-export-destination collector-pic;  # Sends records to the flow collector.
                interface mo-7/1/0.0 {
                    source-address 192.168.252.2;
                }
                interface mo-7/2/0.0 {
                    source-address 192.168.252.2;
                }
                interface mo-7/3/0.0 {
                    source-address 192.168.252.2;
                }
            }
        }
}
firewall {
    family inet {
        filter cp-ftp { # This filter provides CoS for flow collector interface traffic.
            term t1 {
                then forwarding-class expedited-forwarding;
            }
        }
    }
    filter catch { # This firewall filter sends incoming traffic into the
        interface-specific;# filter-based forwarding routing instance.
            term def {
                then {
                    count counter;
                    routing-instance fbf_instance;
                }
            }
        }
    }
routing-options {
    interface-routes {
        rib-group inet common;
    }
    rib-groups {
        common {
            import-rib [inet.0 fbf_instance.inet.0];
```

```
        }
    }
    forwarding-table {
        export pplb;
    }
}
policy-options {
    policy-statement pplb {
        then {
            load-balance per-packet;
        }
    }
}
routing-instances {
    fbf_instance { # This instance sends traffic to the monitoring services interface.
        instance-type forwarding;
        routing-options {
            static {
                route 0.0.0.0/0 next-hop mo-7/1/0.0;
            }
        }
    }
}
class-of-service { # A class-of-service configuration for the flow collector interface
    interfaces { # is required for flow collector services.
        cp-6/0/0 {
            scheduler-map cp-map;
        }
        cp-7/0/0 {
            scheduler-map cp-map;
        }
    }
}
scheduler-maps {
    cp-map {
        forwarding-class best-effort scheduler Q0;
        forwarding-class expedited-forwarding scheduler Q1;
        forwarding-class network-control scheduler Q3;
    }
}
schedulers {
    Q0 {
        transmit-rate remainder;
        buffer-size percent 90;
```

```
    }
    Q1 {
        transmit-rate percent 5;
        buffer-size percent 5;
        priority strict-high;
    }
    Q3 {
        transmit-rate percent 5;
        buffer-size percent 5;
    }
}
services {
    flow-collector { # Define properties for flow collector interfaces here.
        analyzer-address 10.10.10.1; # This is the IP address of the analyzer.
        analyzer-id server1; # This helps to identify the analyzer.
        retry 3; # Maximum number of attempts by the PIC to send a file transfer log.
        retry-delay 30; # The time interval between attempts to send a file transfer log.
        destinations { # This defines the FTP servers that receive flow collector output.
            "ftp://user@192.168.56.89//tmp/collect1/" { # The primary FTP server.
                password "$ABC123"; # SECRET-DATA
            }
            "ftp://user@192.168.252.1//tmp/collect2/" { # The secondary FTP server.
                password "$ABC123"; # SECRET-DATA
            }
        }
        file-specification { # Define sets of flow collector characteristics here.
            def-spec {
                name-format "default-allInt-0-%D_%T-%I_%N.bcp.bi.gz";
                data-format flow-compressed; # The default compressed output format.
            } # When no overrides are specified, a collector uses default transfer values.
            f1 {
                name-format "cFlowd-py69Ni69-0-%D_%T-%I_%N.bcp.bi.gz";
                data-format flow-compressed; # The default compressed output format.
                transfer timeout 1800 record-level 1000000; # Here are configured values.
            }
        }
        interface-map { # Allows you to map interfaces to flow collector interfaces.
            file-specification def-spec; # Flows generated for default traffic are sent to the
            collector cp-7/0/0; # default flow collector interface "cp-7/0/0".
            so-0/1/0.0 { # Flows generated for the so-0/1/0 interface are sent
                collector cp-6/0/0; # to cp-6/0/0, and the file-specification used is
            } # "default."
            so-3/0/0.0 { # Flows generated for the so-3/0/0 interface are sent
                file-specification f1; # to cp-6/0/0, and the file-specification used is "f1."
```

```
            collector cp-6/0/0;
        }
        so-3/1/0.0; # Because no settings are defined, flows generated for this
    } # interface use interface cp-7/0/0 and the default file specification.
    transfer-log-archive { # Sends flow collector interface log files to an FTP server.
        filename-prefix so_3_0_0_log;
        maximum-age 15;
        archive-sites {
            "ftp://user@192.168.56.89//tmp/transfers/" {
                password "$ABC123";
            }
        }
    ]
  }
}
```

RELATED DOCUMENTATION

- Flow Collection Overview | **269**
- Configuring Flow Collection | **270**
- Sending cflowd Records to Flow Collector Interfaces | **282**
- Configuring Flow Collection Mode and Interfaces on Router Services PICs on M and T Series Routers | **283**

Sending cflowd Records to Flow Collector Interfaces

To specify a flow collector interface as the destination for cflowd records coming from a services PIC, include the **collector-pic** statement at the **[edit forwarding-options monitoring *group-name* family inet output flow-export-destination]** hierarchy level:

```
[edit forwarding-options monitoring group-name family inet output flow-export-destination]
collector-pic;
```

You can select either the flow collector interface or a cflowd server as the destination for cflowd records, but not both at the same time.

RELATED DOCUMENTATION

Configuring Flow Collection Mode and Interfaces on Router Services PICs on M and T Series Routers

You can select the services PIC to run in either flow collection mode or monitoring mode, but not both.

To set the services PIC to run in flow collection mode, include the **flow-collector** statement at the **[edit chassis fpc *slot-number* pic *pic-number* monitoring-services application]** hierarchy level:

```
[edit chassis fpc slot-number pic pic-number monitoring-services application]
flow-collector;
```

For further information on configuring chassis properties, see the *Junos OS Administration Library*.

To specify flow collection interfaces, you configure the **cp** interface at the **[edit interfaces]** hierarchy level:

```
[edit interfaces]
cp-fpc/pic/port {
    ...
}
```

RELATED DOCUMENTATION

CHAPTER 5

Logging Flow Monitoring Records with Version 9 and IPFIX Templates for NAT Events

IN THIS CHAPTER

Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250

Starting with Junos OS Release 14.2R2 and 15.1R1, you can configure MX Series routers with MS-MPCs and MS-MICs to log network address translation (NAT) events using the Junos Traffic Vision (previously known as Jflow) version 9 or IPFIX (version 10) template format. You can also configure MX Series routers with MX-SPC3 services cards with this capability starting from Junos OS Release 19.3R2.

NAT event logger generates messages in flow monitoring format for various NAT events, such as the creation of a NAT entry, deletion of a NAT entry, and for invalid NAT processing (such as NAT address pools or address values being exhausted for allocation). These events also support NAT64 translations (translation of IPv6 addresses to IPv4 addresses), binding information base (BIB) events, and more detailed error generation. The generated records or logs for NAT events in flow template format are sent from the MS-MIC or MS-MPC or MX-SPC3 to the specified host or external device that functions as the NetFlow collector. This method of generating flow monitoring records for NAT events enables cohesive and streamlined analysis of NAT traffic and troubleshooting of NAT-related problems. You can enable the capability to send flow monitoring records for NAT operations to an external collector and the capability to use the system logging protocol (syslog) to generate session logging for different services at the same time.

The flow records and the templates are encapsulated in an UDP or IP packet and sent to the collector. However, TCP-based logging of monitoring records for NAT events is not supported. Carrier-grade NAT (CGN) devices are required to log events creation and deletion of translations and information about the resources it manages. Flow monitoring logs can be optionally configured in your network topology in addition to the system logging (syslog) capability, which causes logs to be saved from the PIC to either the in the **/var/log** directory of the Routing Engine (local) or to an external server (remote). Generally, flow collectors are the part of a vast network infrastructure containing several third-party devices, which perform various correlations and mappings with logs of other databases. Therefore, collection of NAT-related flow monitoring records as logs or template records is useful on the hosts or devices that function as collectors in an overall and comprehensive perspective. You can enable logging of flow monitoring records for NAT events at the service-set level to enable version 9 or IPFIX flow records to be generated as logs when NAT is configured on the router.

The NetFlow collector receives flow records in version 9 or IPFIX format from one or more exporters. It processes the received export packets by parsing and saving the flow record details. Flow records can be optionally aggregated before being stored on the hard disk. The NetFlow collector is also referred to as the collector. The exporter monitors packets entering an observation point and creates flows from these packets. The information from these flows is exported in the form of flow records to the NetFlow Collector. An observation point is a location in the network where IP packets can be overseen and monitored; for example, one or a set of interfaces on a network device such as a router. Every observation point is associated with an observation domain, which is a cluster of observation points, and constitutes the largest aggregatable set of flow information at the network device with NetFlow services enabled.

A FlowSet is a generic term for a collection of Flow Records that have a similar pattern or format. In an export packet, one or more FlowSets follow the packet header. A Template FlowSet comprises one or more template records that have been grouped together in an export packet. An Options Template FlowSet contains one or more Options Template records that are combined together in an export packet. A Data FlowSet is one or more records, of the same type, that are grouped together in an export packet. Each record is either a flow data record or an options data record that has been previously specified by a Template Record or an Options Template Record. One of the essential elements in the NetFlow format is the Template FlowSet. Templates vastly enhance the flexibility of the Flow Record format because they allow the collector to process Flow Records without necessarily knowing the interpretation of all the data in the Flow Record.

You can configure the capability to transmit records or log messages in version 9 and IPFIX traffic flow formats generated for NAT events to an external, off-box high-speed NetFlow collector for easy and effective monitoring and diagnosis of the logs. By default, this functionality is disabled. With a high number of NAT events, this mechanism of exporting logs to an external log collector might cause scaling considerations such as loss of a few flow records. To enable the mechanism to record logging messages in flow monitoring format for NAT events, you can now include the **jflow-log** statement at the **[edit services]** hierarchy level. You can configure a collector, which is an external host to which the flow monitoring formatted logs are sent, or a group of collectors. A group of collectors is useful in scenarios in which you want to combine a set of collector devices and define common settings for logging NAT events for all the collectors in the cluster or group.

To configure a collector and its parameters, such as the source IP address from which the records are sent and the destination address of the collector, include the **collector *collector-name*** statement and its substatements at the **[edit services jflow-log]** hierarchy level. To specify a collector group or a cluster, include the **collector-group *collector-group-name*** statement and its substatements at the **[edit services jflow-log]** hierarchy level.

You need to configure a template profile and associate it with the collector. The profile defines the characteristics of the flow monitoring record template, such as the version of flow monitoring (version 9 or IPFIX), the refresh rate, in either packets or seconds, and the type of service or application (NAT in this case) for which flow records must be sent to the collector. To specify a template profile, include the **template-profile *template-profile-name*** statement at the **[edit services jflow-log]** hierarchy level. To specify the maximum number of messages to be collected per second for NAT error events, include the **message-rate-limit *messages-per-second*** statement at the **[edit interfaces *ms-interface-name* service-options jflow-log]** hierarchy level.

Use of version 9 and IPFIX allows you to define a flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic. Templates and the fields included in the template are transmitted to the collector periodically, and the collector need not be aware of the router configuration. You must define a template profile properties for a NAT service and associate the defined template profile with a service set to enable the flow monitoring log functionality for NAT events. To define the template profile characteristics for recording flow monitoring logs for NAT events, include the **template-profile *template-profile-name*** statement at the **[edit services jflow-log]** hierarchy

level. To associate the template profile for recording flow monitoring logs for NAT events with a service-set level, which applies for all the services in the system, include the **template-profile *template-profile-name*** statement at the **[edit services service-set *service-set-name*]** hierarchy level.

To view statistical information on the logs generated in flow monitoring format for the interfaces and service sets configured on the system, use the **show services service-sets statistics jflow-log** command.

The following system log messages for various NAT events are logged using the system logging (syslog) capability:

- JSERVICES_SESSION_OPEN
- JSERVICES_SESSION_CLOSE
- JSERVICES_NAT_OUTOF_ADDRESSES
- JSERVICES_NAT_OUTOF_PORTS
- JSERVICES_NAT_RULE_MATCH
- JSERVICES_NAT_POOL_RELEASE
- JSERVICES_NAT_PORT_BLOCK_ALLOC
- JSERVICES_NAT_PORT_BLOCK_RELEASE
- JSERVICES_NAT_PORT_BLOCK_ACTIVE

The following NAT events are logged using the flow monitoring log capability using version 9 and IPFIX flow templates:

- NAT44 session create
- NAT44 session delete
- NAT addresses exhausted
- NAT64 session create
- NAT64 session delete
- NAT44 BIB create
- NAT44 BIB delete
- NAT64 BIB create
- NAT64 BIB delete
- NAT ports exhausted
- NAT quota exceeded
- NAT Address binding create
- NAT Address binding delete

- NAT port block allocation
- NAT port block release
- NAT port block active

Table 37 on page 288 describes the flow template format for NAT44 session creation and deletion events. The Information Element (IE) names and their IANA IDs are as defined in the IP Flow Information Export (IPFIX) Entities specification by the Internet Assigned Numbering Authority (IANA).

Table 37: Flow Template Format for NAT44 Session Creation and Deletion

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
sourceIPv4Address	32	8
postNATSourceIPv4Address	32	225
protocolIdentifier	8	4
sourceTransportPort	16	7
postNAPTsourceTransportPort	16	227
destinationIPv4Address	32	12
postNATDestinationIPv4Address	32	226
destinationTransportPort	16	11
postNAPTdestinationTransportPort	16	228
natOriginatingAddressRealm	8	229
natEvent	8	230

Table 38 on page 288 describes the flow template format for NAT64 session creation and deletion events.

Table 38: Flow Template Format for NAT64 Session Creation and Deletion

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
sourceIPv6Address	128	27

Table 38: Flow Template Format for NAT64 Session Creation and Deletion *(continued)*

Information Element (IE)	Size (bits)	IANA ID
postNATSourceIPv6Address	32	225
protocolIdentifier	8	4
sourceTransportPort	16	7
postNAPTsourceTransportPort	16	227
destinationIPv6Address	128	28
postNATDestinationIPv6Address	32	226
destinationTransportPort	16	11
postNAPTdestinationTransportPort	16	228
natOriginatingAddressRealm	8	229
natEvent	8	230

Table 39 on page 289 describes the flow template format for NAT44 binding information base (BIB) creation and deletion events.

Table 39: Flow Template Format for NAT44 BIB Creation and Deletion

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
sourceIPv4Address	32	8
postNATSourceIPv4Address	32	225
protocolIdentifier	8	4
sourceTransportPort	16	7
postNAPTsourceTransportPort	16	227
natEvent	8	230

Table 40 on page 290 describes the flow template format for NAT64 binding information base (BIB) creation and deletion events.

Table 40: Flow Template Format for NAT64 BIB Creation and Deletion

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
sourceIPv6Address	128	27
postNATSourceIPv6Address	32	225
protocolIdentifier	8	4
sourceTransportPort	16	7
postNAPTsourceTransportPort	16	227
natEvent	8	230

Table 41 on page 290 describes the flow template format for addresses exhaustion events.

Table 41: Flow Template Format for Address Exhausted Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
natEvent	8	230
natPoolName	512	284

Table 42 on page 290 describes the flow template format for ports exhaustion events.

Table 42: Flow Template Format for Ports Exhausted Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
natEvent	8	230
postNATSourceIPv4Address	32	225
protocolIdentifier	8	4

Table 43 on page 291 describes the flow template format for NAT44 quota exceeded events.

Table 43: Flow Template Format for NAT44 Quota Exceeded Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
natEvent	8	230
sourceIPv4Address	32	8

Table 44 on page 291 describes the flow template format for NAT64 quota exceeded events.

Table 44: Flow Template Format for NAT64 Quota Exceeded Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
natEvent	8	230
sourceIPv6Address	128	27

Table 45 on page 291 describes the flow template format for NAT44 address binding creation and deletion events.

Table 45: Flow Template Format for NAT44 Address Binding Creation and Deletion Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
natEvent	8	230
sourceIPv4Address	32	8
postNATSourceIPv4Address	32	225

Table 46 on page 291 describes the flow template format for NAT64 address binding creation and deletion events.

Table 46: Flow Template Format for NAT64 Address Binding Creation and Deletion Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323

Table 46: Flow Template Format for NAT64 Address Binding Creation and Deletion Events *(continued)*

Information Element (IE)	Size (bits)	IANA ID
natEvent	8	230
sourceIPv6Address	128	27
postNATSourceIPv4Address	32	225

Table 47 on page 292 describes the flow template format for NAT44 port block allocation and deallocation events.

Table 47: Flow Template Format for NAT44 Port Block Allocation and Deallocation Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
sourceIPv4Address	32	8
postNATSourceIPv4Address	32	225
portRangeStart	16	361
portRangeEnd	16	362
portRangeStepSize	16	363
portRangeNumPorts	16	364
observationTimeMilliseconds (time when PBA allocated) **NOTE:** This IE is not included in flow templates when using the MX-SPC3 services card.	64	323
natEvent	8	230

Table 48 on page 293 describes the flow template format for NAT64 port block allocation and deallocation events.

Table 48: Flow Template Format for NAT64 Port Block Allocation and Deallocation Events

Information Element (IE)	Size (bits)	IANA ID
observationTimeMilliseconds	64	323
sourceIPv6Address	128	27
postNATSourceIPv4Address	32	225
portRangeStart	16	361
portRangeEnd	16	362
portRangeStepSize	16	363
portRangeNumPorts	16	364
observationTimeMilliseconds (time when port block allocation (PBA) is configured) **NOTE:** This IE is not included in flow templates when using the MX-SPC3 services card.	64	323
natEvent	8	230

In all of the aforementioned templates, the natEvent field maps to one of the values listed in Table 49 on page 293, depending on the type of event.

Table 49: Association Between natEvent Values and Names

natEvent Value	natEvent Name
1	NAT44 Session create
2	NAT44 Session delete
3	NAT Addresses exhausted
4	NAT64 Session create
5	NAT64 Session delete
6	NAT44 BIB create

Table 49: Association Between natEvent Values and Names *(continued)*

natEvent Value	natEvent Name
7	NAT44 BIB delete
8	NAT64 BIB create
9	NAT64 BIB delete
10	NAT ports exhausted
11	NAT Quota exceeded
12	NAT Address binding create
13	NAT Address binding delete
14	NAT port block allocation
15	NAT port block release
16	NAT port block active

Release History Table

Release	Description
19.3R2	You can also configure MX Series routers with MX-SPC3 services cards with this capability starting from Junos OS Release 19.3R2.

RELATED DOCUMENTATION

Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**

Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**

Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250

Keep the following points in mind when you configure the capability to generate logs or records in flow monitoring format for NAT events:

- Enabling syslog and Jflow capabilities at the same time might result in scaling impacts because both these mechanisms use a separate infrastructure to transfer records out to the collector.
- High number of NAT events can cause scalability considerations because of the flow monitoring framework too requiring system processes.
- The flow monitoring log infrastructure uses data CPUs to send the logs to the external flow server, which might cause a slight impact on performance.
- An explicit, separate maximum limit on the number of flow monitoring messages that are generated for NAT error events is implemented. You can control the maximum number of NAT error events for which logs in flow monitoring format must be recorded by including the **message-rate-limit *messages-per-second*** option at the **[edit interfaces *interface-name* services-options jflow-log]** hierarchy level. These records for NAT error events are generated when addresses for allocation from the NAT pool are not available, when ports for allocation to a subscriber are not available, or when the allocated quota is exceeded for NAT events (more than the configured number of ports is requested). Also, you can configure the **message-rate-limit** option that previously existed at the **[edit interfaces *interface-name* services-options syslog]** hierarchy level to specify the maximum number of system log messages per second that can be sent from the PIC to either the Routing Engine (local) or to an external server (remote).
- NAT error events such as “Out of Ports”, “Out of Addresses” and “Quota Exceeded” are rate limited. Default rate limit is 10,000 events per second. This setting is also configurable at PIC level.
- The template for NAT event logging is in accordance with IETF as *IPFIX Information Elements for logging NAT Events—draft-ietf-behave-ipfix-nat-logging-02*.
- Only UDP-based logging is supported, which is an unreliable protocol.
- This functionality is supported on MX Series routers with Junos OS Extension-Provider packages installed and configured on the device, and on MS-MPCs, MS-PICs, and MX-SPC3s. It is not supported on MS-DPCs with MX Series routers.
- Transmission of logs occurs in clear-text format similar to other log messages that the services PICs do not encrypt. It is assumed that the transport of logs and the positioning of the log collector are within a secured realm. Because the messages do not contain sensitive details such as username or passwords, the messages do not cause any security or reliability risks.
- Template IDs 0 through 255 are reserved for template sets and the maximum number of templates supported for logging events in flow monitoring format is 255. When you modify a template-profile configuration (changes to the collector or version, or a deactivation and activation of the service set associated with the template), the specific template is deregistered and reregistered. However, the flow monitoring infrastructure requires 10 minutes by default as the delay period for the template IDs to be

freed up. As a result, if you modify the template-profile settings many times within the 10-minute period, the maximum limit on the template IDs of 255 is exceeded and further templates are not registered. In such a case, when templates are not being registered, you must wait until the delay period for deleting a deregistered template of 10 minutes before you perform any more configuration changes to have templates registered with the flow monitoring application. To examine whether a template has been registered, you can use the **show services service-sets statistics jflow-log** command. If the Sent field displays a non-zero value for template records, it denotes that templates are successfully registered.

- In a scenario in which the capability to log NAT events in flow monitoring format is enabled at the service-set level, and if the PIC boots up, the flow monitoring log templates are registered with the flow monitoring application. During the registration process, a first set of 12 template records are sent to the collector. However, all of the template records might not reach the collector from the PIC on the router or might not be transmitted out of the router because the interface might not be up from the perspective of the Packet Forwarding Engine. After the refresh time of a template expires, next set of template records are sent out to the collector. For example, if the template refresh time is 60 seconds, only after 60 seconds from the time of booting of the PIC, template records are properly sent to the collector.
- If no problems occur in the transmission of flow monitoring log messages to the collector from the PIC, the Sent field is incremented to indicate NAT events being logged for every event. Also, the tcpdump utility at the destination IP address of the collector denotes the reception of UDP packets. If NAT processing occurs and the value in the **Dropped** section of the output of the **show services service-sets statistics jflow-log service-set *service-set-name*** command is incremented or not incremented, you must examine the debugging statistics and counters to determine if any problems exist in the network for transmission of the flow monitoring log messages.

RELATED DOCUMENTATION

Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**

Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**

Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

Exporting Syslog Messages to an External Host Without Flow Monitoring Formats Using an MX Series Router or NFX250

Until Junos OS Release 14.2R1, the only mechanism you can use to generate logs for NAT sessions was by enabling system logging for service sets and transferring syslog messages to either the internal local host on the Routing Engine or to an external host server. When a syslog is enabled with the class or component being NAT logs and session logs configured, NAT events are recorded. A sample of one such syslog output is as follows:

```
{service_set_3}[jservices-nat]: JSERVICES_NAT_RULE_MATCH: proto 17(UDP) app: any,
 xe-3/1/1.0#012 192.0.2.2/18575 -> 23.0.0.2/63,Match NAT rule-set (null) rule
nat-basic_1
term t1
{service_set_3}MSVCS_LOG_SESSION_OPEN: App:none, xe-3/1/1.0#012 24.0.0.2:18575
[198.51.100.17:1048] -> 23.0.0.2:63  (UDP)
{service_set_3}MSVCS_LOG_SESSION_CLOSE: App:none, xe-3/1/1.0#012 24.0.0.2:18575
[198.51.100.17:1048] -> 23.0.0.2:63  (UDP)
```

From the preceding syslog output, it denotes that NAT create log (NAT translation) and delete log (NAT release) are generated during session events as a part of session-logs configuration. Another important log that is NAT pool exhaustion (not illustrated in the preceding example) is generated as a part of NAT-logs configuration. Such an event message might be caused by Address pooling paired (APP), endpoint-independent mapping (EIM), or address and port exhaustion.

Exporting Version 9 Flow Data Records to a Log Collector Overview Using an MX Series Router or NFX250

A flow record template defines a collection of fields with corresponding descriptions of the format and syntax for the elements or attributes that are contained in it. Network elements (such as routers and switches), which are called exports, accumulate the flow data and export the information to collectors, which are hosts or external devices that can save a large volume of such system log messages for events or system operations. The collected data provides granular, finer-level metering and statistical data for highly flexible and detailed resource usage accounting. Templates that are sent to the collector contain the structural information about the exported flow record fields; therefore, if the collector cannot interpret the formats of the new fields, it can still process the flow record.

The version 9 flow template has a predefined format. An export packet consists of a packet header followed by one or more FlowSet fields. The FlowSet fields can be any of the possible three types—Template, Data,

or Options Template. The template flowset describes the fields that are in the data flowsets (or flow records). Each data flowset contains the values or statistics of one or more flows with the same template ID. An interleaved NetFlow version 9 export packet contains the packet header, Template FlowSet, and Data FlowSet fields. A Template FlowSet field signifies each event such as the creation of a NAT entry or the release of a NAT entry allocated, and the Data FlowSet field denotes the NAT sessions for which the Template FlowSet (or the event type) is associated. For example, if a NAT address entry creation, exhaustion of addresses in a NAT pool, and a NAT entry deletion or release occur, an interleaved version 9 export packet contains the packet header, one Template FlowSet field for NAT address creation, two Data FlowSet fields for the two sessions for which address creation is performed, another TemplateSet field for NAT address deletion, two Data FlowSet fields for the two sessions for which address deletion event occurs, and the other TemplateSet field for NAT pool consumption having exceeded the configured number of pools.

The following are the possible combinations that can occur in an export packet:

- An export packet that consists of interleaved template and data FlowSets—A collector device should not assume that the template IDs defined in such a packet have any specific relationship to the data FlowSets within the same packet. The collector must always cache any received templates, and examine the template cache to determine the appropriate template ID to interpret a data record.
- An export packet consisting entirely of data FlowSets—After the appropriate template IDs have been defined and transmitted to the collector device, most of the export packets consist solely of data FlowSets.
- An export packet consisting entirely of template FlowSets—Although this case is the exception, it is possible to receive packets containing only template records. Ordinarily, templates are appended to data FlowSets. However, in some instances only templates are sent. When a router first boots up or reboots, it attempts to synchronize with the collector device as quickly as possible. The router can send template FlowSets at an accelerated rate so that the collector device has sufficient information to parse any subsequent data FlowSets. Also, template records have a limited lifetime, and they must be periodically refreshed. If the refresh interval for a template occurs and no appropriate data FlowSet that needs to be sent to the collector device is present, an export packet consisting only of template FlowSets is sent.

Understanding Exporting IPFIX Flow Data Records to a Log Collector Using an MX Series Router or NFX250

The IPFIX protocol enables you to access IP flow information on MX-Series Routers or an NFX250. The IPFIX collection process receives the flow information traversing through multiple network elements within the data network in a consistent, identical manner of representation and communication of traffic flows from the network elements to the collection point. An IPFIX device hosts at least one exporting process, which transmits flow records to collecting processes. A collector is a device that performs the collecting processes and an exporter is a device that performs the transfer to data to a collector. An IPFIX nessage consists of a message header followed by one or more Sets. The Sets can be any of the possible three types: Data Set, Template Set, or Options Template Set. Flow monitoring version 10 (IPFIX) message formats are very similar to version 9 message patterns.

The message header contains the following fields:

- Version—Version of the flow record format exported in this message. The value of this field is 0x000a.
- Length—Total length of the IPFIX message, measured in octets, including the header and Sets fields.
- Export Time—Time, in seconds, since midnight Coordinated Universal Time (UTC) of January 1, 1970, at which the IPFIX message header leaves the exporter. Sequence Number—Incremental sequence counter with a value of 2^32 (2 raised to the power of 32) of all IPFIX data records sent from the current Observation Domain by the exporting process. Template and Options Template records do not increase the Sequence Number attribute.
- Observation Domain ID—A 32-bit identifier of the Observation Domain that is locally unique to the exporter.

One of the essential elements in the IPFIX record format is the Template FlowSet record. Templates vastly enhance the flexibility of the Flow Record format because they allow the collector to process Flow Records without necessarily knowing the interpretation of all the data in the Flow Record. A Template Record contains any combination of Internet Assigned Numbers Authority (IANA)-assigned and/or enterprise-specific information element identifiers.

The format of the Template Record signifies a template record header and one or more Field Specifier attributes. The Template FlowSet record contains the following fields:

- Enterprise bit—This is the first bit of the Field Specifier. If this bit is zero, the Information Element Identifier identifies an IETF-specified Information Element, and the four-octet Enterprise Number field must not be present. If this bit is one, the Information Element identifier identifies an enterprise-specific Information Element, and the Enterprise Number field must be present.
- Information Element identifier—An Information Element is a protocol and encoding-independent description of an attribute that can appear in an IPFIX Record. It is a numeric value that represents the type of Information Element.

- Field Length—Length of the corresponding encoded Information Element, in octets. The value 65535 is reserved for variable-length Information Elements.
- Enterprise Number—IANA enterprise number of the authority defining the Information Element identifier in this Template Record.

The Data Records are sent in Data Sets. The Data Record field consists only of a Set Header and one or more Field Values. The Template ID to which the Field Values belong is encoded in the Set Header field "Set ID" ("Set ID" = "Template ID"). Interpretation of the Data Record format can be done only if the Template Record corresponding to the Template ID is available at the collecting procedure. Field Values do not necessarily have a length of 16 bits and are encoded according to their data type specified.

Mapping Between Field Values for Version 9 Flow Templates and Logs Exported From an MX-Series Router or NFX250

The following table describes different field IDs or values for flow monitoring logs generated for NAT events in version 9 flow record formats and the events that correspond to the field values:

Field ID	Name	Size (Bytes)	Description
8	ipv4 src address	4	IPv4 source address
225	natInsideGlobalAddress	4	It reports a modified value caused by a NAT middlebox (forwarding class and loss priority) represents function after the packet passed the Observation Point.
12	ipv4 destination address	4	IPv4 destination address
226	natOutsideGlobalAddress	4	It reports a modified value caused by a NAT middlebox function after the packet passed the Observation Point.
7	transport source-port	2	TCP/UDP source port
227	postNAPTSourceTransportPort	2	It reports a modified value caused by a Network Address Port Translation (NAPT) middlebox function after the packet passed the Observation Point.
11	transport destination-port	2	TCP/UDP destination port
228	postNAPTDestinationTransportPort	2	It reports a modified value caused by a Network Address Port Translation (NAPT) middlebox function after the packet passed the Observation Point.

Field ID	Name	Size (Bytes)	Description
234	ingressVRFID	4	Unique identifier of the VRF name where the packets of this flow are being received. This identifier is unique per Metering Process.
235	egressVRFID	4	Unique identifier of the VRF name where the packets of this flow are being sent. This identifier is unique per Metering Process.
4	Ip protocol	1	IP protocol byte
229	natOriginatingAddressRealm	1	Indicates whether the session was created because traffic originated in the private or public address realm. postNATSourceIPv4Address, postNATDestinationIPv4Address, postNAPTSourceTransportPort, and postNAPTDestinationTransportPort are qualified with the address realm in perspective. The allowed values are: Private: 1 Public: 2
230	natEvent	1	Indicates a NAT event. The allowed values are: 1 - Create event. 2 - Delete event. 3 - Pool exhausted. A Create event is generated when a NAT translation is created, whether dynamically or statically. A Delete event is generated when a NAT translation is deleted.
1	inBytes	N	Incoming counter with length N x 8 bits for the number of bytes associated with an IP Flow. By default N is 4
2	inPkts	N	Incoming counter with length N x 8 bits for the number of packets associated with an IP Flow. By default N is 4
323	observationTimeMilliseconds	8	Specifies the absolute time in milliseconds of an observation that represents a time value in units of milliseconds based on coordinated universal time (UTC). The choice of an epoch, for example, 00:00 UTC, January 1, 1970, is left to corresponding encoding specifications for this type. Leap seconds are excluded. Note that transformation of values might be required between different encodings if different epoch values are used.

Field ID	Name	Size (Bytes)	Description
27	sourceIPv6Address	16	IPv6 source address
284	natPoolName	64	NAT resource pool name
361	portRangeStart	2	The port number identifying the start of a range of ports. A value of zero indicates that the range start is not specified, ie the range is defined in some other way.
362	portRangeEnd	2	The port number identifying the end of a range of ports. A value of zero indicates that the range end is not specified, and the range is defined in some other way.
363	portRangeStepSize	2	The step size in a port range. The default step size is 1, which indicates contiguous ports. A value of zero indicates that the step size is not specified, and the range is defined in some other way.
364	portRangeNumPorts	2	The number of ports in a port range. A value of zero indicates that the number of ports is not specified, and the range is defined in some other way.

Consider a sample scenario of a NAT address creation event. Based on the fields in the preceding table, for translations that are not available (such as natOutsideGlobalAddress) is set to 0. Ingress and Egress VRF of the flow can be made available. Also, natEvent is equal to 1 (create). The inBytes field is assumed to be 0 or number of bytes of the incoming packet and the inPkts field is either 0 or 1 because it is the first packet into the system when translation happens. The observationTimeMilliseconds field denotes the time when this address translation creation is recorded.

For a NAT address deletion event, for translations that are not available (such as natOutsideGlobalAddress) is set to 0. Ingress and Egress VRF of the flow can be made available. Also, natEvent is equal to 2 (create). The inBytes field denotes the number of bytes for this flow in both the forward or upward, the value of the inPkts field denotes the number of packets for this flow in both the upward and backward directions. observationTimeMilliseconds is the time when this deletion of translation is recorded.

When the NAT pool is exhausted and no further addresses are remaining for allocation, for translations that are not available (such as natOutsideGlobalAddress) is set to 0. Ingress and Egress VRF of the flow can be made available. Also, the natEvent field is set to 3 (Pool exhausted). All resource failures are combined as a single event. The inBytes field is assumed to be 0 or number of bytes of the incoming packet and the inPkts field is either 0 or 1 because it is the first packet into the system when translation happens. The value of the observationTimeMilliseconds field is the time when this failed translation is recorded.

Mapping Between Field Values for IPFIX Flow Templates and Logs Exported From an MX Series Router or NFX250

A new proposed draft defining IPFIX IEs for logging various NAT events is available in IETF as *IPFIX Information Elements for logging NAT Events—draft-ietf-behave-ipfix-nat-logging-02*. The flow monitoring template format for flow monitoring logs generated for NAT events comply with the the templates defined in this draft for logging NAT44/NAT64 session create/delete, binding information base (BIB) create/delete, address exhaust, pool exhaustion, quota exceeded, address binding create/delete, port block allocation and de-allocation events. Also, this draft has an extension for NAT64. Support is implemented for logging events for both NAT44 and NAT64. Apart from those templates defined in this draft, no new user-defined templates are created for logging any NAT events.

The following table lists the extensions to the NAT events. The data record contains the corresponding natEvent value to identify the event that is being logged.

Event Name	Values
NAT44 Session create	1
NAT44 Session delete	2
NAT Addresses exhausted	3
NAT64 Session create	4
NAT64 Session delete	5
NAT44 BIB create	6
NAT44 BIB delete	7
NAT64 BIB create	8
NAT64 BIB delete	9
NAT ports exhausted	10
Quota exceeded	11
Address binding create	12
Address binding delete	13
Port block allocation	14

Event Name	Values
Port block deallocation	15

The following table describes the field IDs or values and the corresponding names for IPv6 addresses for IPFIX flows:

Field ID	Name	Size (Bytes)	Description
27	sourceIPv6Address	16	IPv6 source address
28	destinationIPv6Address	16	IPv6 destination address
281	postNATSourceIPv6Address	16	Translated source IPv6 address
282	postNATDestinationPv6Address	16	Translated destination IPv6 address

The following table describes the field names and whether they are required or not for NAT64 session creation and deletion events:

Field Name	Size (Bytes)	Whether the Field Is Mandatory
timeStamp	64	Yes
vlanID/ingressVRFID	32	No
sourceIPv4Address	128	Yes
postNATSourceIPv4Address	32	Yes
protocolIdentifier	8	Yes
sourceTransportPort	16	Yes
postNAPTsourceTransportPort	16	Yes
destinationIPv4Address	128	No
postNATDestinationIPv4Address	32	No
destinationTransportPort	16	No
postNAPTdestinationTransportPort	16	No

Field Name	Size (Bytes)	Whether the Field Is Mandatory
natOriginatingAddressRealm	8	No
natEvent	8	Yes

A NAT44 session creation template record can contain the following fields. The natEvent field contains a value of 1, which indicates a NAT44 session creation event. An example of such a template is as follows:

Field Name	Size (Bytes)	Value
timeStamp	64	09:20:10:789
sourceIPv4Address	32	192.168.16.1
postNATSourceIPv4Address	32	192.0.2.100
protocolIdentifier	8	TC
sourceTransportPort	16	14800
postNAPTsourceTransportPort	16	1024
destinationIPv4Address	32	198.51.100.104
postNATDestinationIPv4Address	32	198.51.100.104
destinationTransportPort	16	80
postNAPTdestinationTransportPort	16	80
natOriginatingAddressRealm	8	0
natEvent	8	1

A NAT44 session deletion template record can contain the following fields. The natEvent field contains a value of 2, which indicates a NAT44 session deletion event. An example of such a template is as follows:

Field Name	Size (Bytes)	Value
timeStamp	64	09:20:10:789
sourceIPv4Address	32	192.168.16.1

Field Name	Size (Bytes)	Value
postNATSourceIPv4Address	32	192.0.2.100
protocolIdentifier	8	TC
sourceTransportPort	16	14800
postNAPTsourceTransportPort	16	1024
destinationIPv4Address	32	198.51.100.104
postNATDestinationIPv4Address	32	198.51.100.104
destinationTransportPort	16	80
postNAPTdestinationTransportPort	16	80
natOriginatingAddressRealm	8	0
natEvent	8	2

Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats

You can configure MX Series routers with MS-MPCs, MS-MICs, and MX-SPC3s to log network address translation (NAT) events using the Junos Traffic Vision (previously known as Jflow) version 9 or IPFIX (version 10) template format. NAT event logger generates logs or template records in flow monitoring format and transmits them to the specified external collector or server for various NAT events, such as NAT44 and NAT64 session creation and deletion, and NAT44 and NAT64 binding information base events.

> **NOTE:** This functionality is supported on MX Series routers with Junos OS Extension-Provider packages installed and configured on the device, and on MS-MPCs, MS-PICs, and MX-SPC3s. It is not supported on MS-DPCs with MX Series routers.

You can configure the mechanism to record logging messages in flow monitoring format for NAT events. You need to define collectors, and template profiles that contain the properties for flow monitoring logs. You can create a template profile for a particular NAT service on an MX Series router with MS-MPCs, MS-MICs, or MX-SPC3s, or for a service set, which applies for all of the NAT services. You can define a template profile to generate flow monitoring logs in a specific flow template format and associate the specified template profile with a service set.

To enable the flow monitoring log capability for NAT events and configure the transmission of logs to collectors at a service level:

1. Define the flow monitoring log service to be applied on an interface to control the maximum number of flow monitoring logs generated for NAT error events.

   ```
   [edit]
   user@host# set interfaces ms-fpc/pic/port services-options jflow-log message-rate-limit messages-per-second
   ```

   For example:

   ```
   [edit]
   user@host# set interfaces ms-5/0/0 services-options jflow-log message-rate-limit 50
   ```

2. Configure the collectors and collector groups.

   ```
   [edit]
   user@host# set services jflow-log collector collector-name destination-address address destination-port
     port-number source-ip address
   ```

```
user@host# set services jflow-log collector-group collector-group-name collector [ collector-name1
    collector-name2]
```

For example:

```
[edit]
user@host# set services jflow-log collector c1 destination-address 203.0.113.3 destination-port 1 source-ip
    192.0.2.1
user@host# set services jflow-log collector-group cg1 collector c1
```

3. Configure the template profiles and associate the template profile with the collector or collector group.

```
[edit]
user@host# set services jflow-log template-profile template-profile-name collector collector-name version
    (ipfix | v9) template-type nat refresh-rate packets packets seconds seconds
user@host# set services jflow-log template-profile template-profile-name collector-group collector-group-name
    version (ipfix | v9) template-type nat refresh-rate packets packets seconds seconds
```

For example:

```
[edit]
user@host# set services jflow-log template-profile t1 collector c1 version ipfix template-type nat refresh-rate
    packets 20 seconds 20
user@host# set services jflow-log template-profile t1 collector-group cg1
user@host# set services jflow-log template-profile t2 collector c2 version v9 template-type nat refresh-rate
    packets 20 seconds 20
```

4. Associate the template profile with the service set.

```
[edit]
user@host# set services service-set service-set-name jflow-log template-profile template-profile-name
```

For example:

```
[edit]
user@host# set services service-set sset_0 jflow-log template-profile t1
```

RELATED DOCUMENTATION

Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting

IN THIS SECTION

You can configure MX Series routers with MS-MPCs, MS-MICs, and MX-SPC3s to log network address translation (NAT) events using the Junos Traffic Vision (previously known as Jflow) version 9 or IPFIX (version 10) template format. This method of generating flow monitoring records for NAT events, such as NAT44 and NAT64 session creation and deletion, and NAT44 and NAT64 binding information base events, enables cohesive and streamlined analysis of NAT traffic and troubleshooting of NAT-related problems.

NOTE: This functionality is supported on MX Series routers with Junos OS Extension-Provider packages installed and configured on the device, and on MS-MPCs, MS-PICs, and MX-SPC3s. It is not supported on MS-DPCs with MX Series routers.

This example describes how to configure flow monitoring log generation in flow monitoring format for NAT events at the service-set level on MS-MIC, MS-MPC, and MX-SPC3, and contains the following sections:

NOTE: This configuration example is for an Interface-Style service set.

Requirements

This example uses the following hardware and software components:

- One MX Series router with an MS-MPC, MS-MIC, or MX-SPC3
- Junos OS Release 14.2R2 or later for MX Series routers

Generation of Log Messages Using Flow Templates for NAT Operations on MS-MPCs, MS-MICs, and MX-SPC3s

You can configure the mechanism to record logging messages in flow monitoring format for NAT events. You can create a template profile for a particular NAT service on an MX Series router with MS-MPCs, MS-MICs, or MX-SPC3s, or for a service set, which applies for all of the NAT services. You must define a template profile to generate flow monitoring logs in a specific flow template format and attach the template profile with a service set. You must configure a collector or a group of collectors, which are hosts that receive the log messages for NAT events from the service PIC or the exporter. You need to associate a template profile with the collector. The profile defines the characteristics of the flow monitoring record template, such as the version of flow monitoring (version 9 or IPFIX), the refresh rate, in either packets or seconds, and the type of service or application (NAT in this case) for which flow records must be sent to the collector.

Assume a sample deployment in which two collectors, c1 and c2, are defined. These collectors are clustered into two groups. The collector group, cg1, contains c1 and c2, and the collector group, cg2, contains c2. Two template profiles named t1 and t2 are defined. The profiles, t1 and t2, are associated with collectors, c1 and c2, respectively.

These profiles describe the properties or attributes for transmission of logs, such as the flow template format to be used, the rate at which the logs must be refreshed, and the service or event, such as NAT, for which logs must be sent to the specified collector.

Configuration

To enable the flow monitoring log capability for NAT events and configure the transmission of logs to collectors, 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:

Configuring Service Set Properties

```
set services service-set sset_0 interface-service service-interface ms-5/0/0.0
```

Applying Flow Monitoring Log Service on an Interface

```
set interfaces ms-5/0/0 services-options jflow-log message-rate-limit 50000
```

Enabling and Configuring Flow Monitoring Logs for a Service Set

```
set services jflow-log collector c1 destination-address 192.0.2.3 destination-port 1 source-ip
   198.51.100.1
set services jflow-log collector c2 destination-address 203.0.113.5 destination-port 3 source-ip
   198.51.100.2
set services jflow-log collector-group cg1 collector [ c1 c2 ]
set services jflow-log template-profile t1 collector c1 version ipfix template-type nat refresh-rate
   packets 20 seconds 20
set services jflow-log template-profile t2 collector c2 version v9 template-type nat refresh-rate packets
   20 seconds 20
set services jflow-log template-profile t1 collector-group cg1
```

Associating the Template Profile with a Service Set

```
set services service-set sset_0 jflow-log template-profile t1
```

Step-by-Step Procedure

To configure the generation and transmission of flow monitoring template logs for NAT events:

1. Create a service set properties.

   ```
   [edit]
   user@host# set services service-set sset_0 interface-service service-interface ms-5/0/0.0
   ```

2. Define the flow monitoring log service to be applied on an interface.

```
[edit]
user@host# set interfaces ms-5/0/0 services-options jflow-log message-rate-limit 50000
```

3. Configure the collectors and collector groups.

```
[edit]
user@host# set services jflow-log collector c1 destination-address 192.0.2.3 destination-port 1 source-ip
   198.51.100.1
user@host# set services jflow-log collector c2 destination-address 203.0.113.5 destination-port 3 source-ip
   198.51.100.2
user@host# set services jflow-log collector-group cg1 collector [ c1 c2 ]
user@host# set services jflow-log collector-group cg2 collector c2
```

4. Configure the template profiles and associate the template profile with the collector.

```
[edit]
user@host# set services jflow-log template-profile t1 collector c1 version ipfix template-type nat refresh-rate
   packets 20 seconds 20
user@host# set services jflow-log template-profile t2 collector c2 version v9 template-type nat refresh-rate
   packets 20 seconds 20
```

5. Associate the template profile with the service set.

```
[edit]
user @ host# set services service-set sset_0 jflow-log template-profile t1
```

Results

From the configuration mode, confirm your configuration by entering the **show services**, **show services jflow-log**, and **show services service-set sset_0 jflow-log** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show services
service-set sset_0 {
   interface-service {
      service-interface ms-5/0/0;
   }
}
```

```
[edit interfaces]
ms-5/0/0 {
  services-options {
   jflow-log {
    message-rate-limit 50000;
   }
  }
}
```

```
user@host# show services jflow-log
collector c1 {
    destination-address 192.0.2.3;
    destination-port 1;
    source-ip 198.51.100.1;
}
collector c2 {
    destination-address 203.0.113.5;
    destination-port 3;
    source-ip 198.51.100.2;
}
collector-group cg1 {
    collector [ c2 c1 ];
}
collector-group cg2 {
    collector c2;
}
template-profile t2 {
    collector c2;
    template-type nat;
    referesh-rate packets 20 seconds 20;
    version v9;
}
template-profile t1 {
    collector c1;
    template-type nat;
    referesh-rate packets 20 seconds 20;
    version ipfix;
}
```

```
[edit]
user@host# show services service-set sset_0 jflow-log
template-profile t2;
```

Verification

IN THIS SECTION

To confirm that the configuration is working properly, perform the following:

Verifying That the Flow Monitoring Logs Are Generated and Sent to Collectors

Purpose

Verify that the flow monitoring log messages in the defined template format, such as IPFIX or version 9, are generated and transmitted to the configured collectors for the different NAT operations.

Action

From operational mode, use the **show services service-sets statistics jflow-log** command:

`user@host>` **show services service-sets statistics jflow-log**

```
Interface: ms-5/0/0
  Rate limit: 1000
  Template records:
    Sent: 36
    Dropped: 0
  Data records:
    Sent: 2
    Dropped: 0

    Service-set: sset_0
      Unresolvable collectors: 0
      Template records:
        Sent: 36
        Dropped: 0
      Data records:
        Sent: 2
```

```
        Dropped: 0
```

From operational mode, use the **show services service-sets statistics jflow-log detail** command:

`user@host>` **show services service-sets statistics jflow-log detail**

```
Interface: ms-5/0/0
  Rate limit: 1000
  Template records:
    Sent: 48
    Dropped: 0
  Data records:
    Sent: 4
    Dropped: 0

    Service-set: sset_0
      Unresolvable collectors: 0
      Template records:
        Sent: 48
        Dropped: 0
      Data records:
        Sent: 4
        Dropped: 0
      NAT44 Session logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 4
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT64 Session logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 0
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT44 BIB logs:
        Template records:
```

```
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 BIB logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT Address Exhausted logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT Port Exhausted logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 Quota Exceeded logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 Quota Exceeded logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 Address Bind logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
```

```
    Data records:
      Sent: 0
      Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
  NAT64 Address Bind logs:
    Template records:
      Sent: 4
      Dropped: 0 (socket send error: 0, no memory: 0)
    Data records:
      Sent: 0
      Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
  NAT44 PBA logs:
    Template records:
      Sent: 4
      Dropped: 0 (socket send error: 0, no memory: 0)
    Data records:
      Sent: 0
      Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
  NAT64 PBA logs:
    Template records:
      Sent: 4
      Dropped: 0 (socket send error: 0, no memory: 0)
    Data records:
      Sent: 0
      Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
```

Meaning

The output shows that the log messages in flow monitoring format associated with the specified service set and interface are generated for the different NAT events.

RELATED DOCUMENTATION

Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**

Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**

Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**

2

PART

Flow Capture Services

CHAPTER 6

Dynamically Capturing Packet Flows Using Junos Capture Vision

IN THIS CHAPTER

Understanding Junos Capture Vision

IN THIS SECTION

Junos Capture Vision (known as dynamic flow capture in Junos OS Releases earlier than 13.2) enables you to capture packet flows on the basis of dynamic filtering criteria. Specifically, you can use this feature to forward passively monitored packet flows that match a particular filter list to one or more destinations using an on-demand control protocol.

This topic contains the following sections:

Junos Capture Vision Architecture

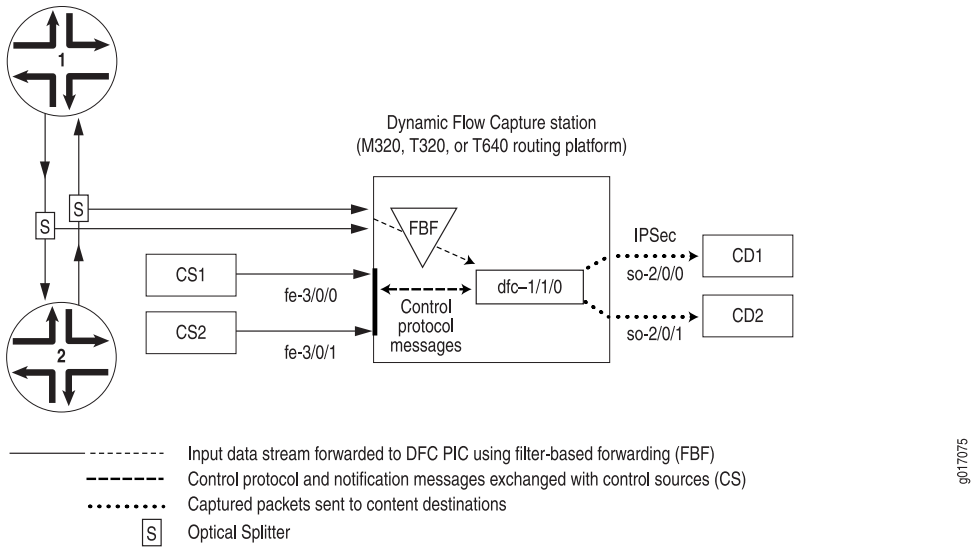
The architecture consists of one or more *control sources* that send requests to a Juniper Networks router to monitor incoming data, and then forward any packets that match specific filter criteria to a set of one or more *content destinations*. The architectural components are defined as follows:

- Control source—A client that monitors electronic data or voice transfer over the network. The control source sends filter requests to the Juniper Networks router using the Dynamic Task Control Protocol (DTCP), specified in draft-cavuto-dtcp-03.txt at http://www.ietf.org/internet-drafts. The control source is identified by a unique identifier and an optional list of IP addresses.
- Monitoring platform—A T Series or M320 router containing one or more Dynamic Flow Capture (DFC) PICs, which support dynamic flow capture processing. The monitoring platform processes the requests from the control sources, creates the filters, monitors incoming data flows, and sends the matched packets to the appropriate content destinations.
- Content destination—Recipient of the matched packets from the monitoring platform. Typically the matched packets are sent using an IP Security (IPsec) tunnel from the monitoring platform to another router connected to the content destination. The content destination and the control source can be physically located on the same host. For more information on IPsec tunnels, see *Understanding Junos VPN Site Secure*.

> **NOTE:** The Junos Capture Vision PIC (either a Monitoring Services III PIC or Multiservices 400 PIC) forwards the entire packet content to the content destination, rather than to a content record as is done with cflowd or flow aggregation version 9 templates.

Figure 30 on page 320 shows a sample topology. The number of control sources and content destinations is arbitrary.

Figure 30: Junos Capture Vision Topology



Liberal Sequence Windowing

Each DTCP packet (add, delete, list, and refresh packets) contains a 64-bit sequence number to identify the order of the packets. Because the network is connectionless, the DTCP packets can arrive out of order to the router running the Junos Capture Vision application.

The *liberal sequence window* feature implements a negative window for the sequence numbers received in the DTCP packets. It enables the Junos Capture Vision application to accept not only DTCP packets with sequence numbers greater than those previously received, but also DTCP packets with lesser sequence numbers, up to a certain limit. This limit is the negative window size; the positive and negative window sizes are +256 and −256 respectively, relative to the current maximum sequence number received. No configuration is required to activate this feature; the window sizes are hard-coded and nonconfigurable.

Intercepting IPv6 Flows

Starting with Junos OS Release 11.4, Junos Capture Vision also supports intercepting IPv6 flows in M320, T320, T640, and T1600 routers with a Multiservices 400 or Multiservices 500 PIC. Junos Capture Vision can intercept passively monitored IPv6 traffic only. All support for IPv4 interception remains the same. The interception of IPv6 traffic happens in the same way the filters capture IPv4 flows. With the introduction of IPv6 interception, both IPv4 and IPv6 filters can coexist. The mediation device, however, cannot be located in an IPv6 network.

Junos Capture Vision does not support interception of VPLS and MPLS traffic. The application cannot intercept Address Resolution Protocol (ARP) or other Layer 2 exception packets. The interception filter can be configured to timeout based on factors like total time (seconds), idle time (seconds), total packets or total data transmitted (bytes).

RELATED DOCUMENTATION

Configuring Junos Capture Vision | 321

Example: Configuring Junos Capture Vision on M and T Series Routers | 329

Configuring Junos Capture Vision

IN THIS SECTION

This section describes the following tasks for configuring Junos Capture Vision:

Configuring the Capture Group

A capture group defines a profile of Junos Capture Vision configuration information. The static configuration includes information about control sources, content destinations, and notification destinations. Dynamic configuration is added through interaction with control sources using a control protocol.

To configure a capture group, include the **capture-group** statement at the **[edit services dynamic-flow-capture]** hierarchy level:

```
capture-group client-name {
    content-destination identifier {
        address address;
        hard-limit bandwidth;
        hard-limit-target bandwidth;
        soft-limit bandwidth;
        soft-limit-clear bandwidth;
        ttl hops;
    }
    control-source identifier {
        allowed-destinations [ destinations ];
        minimum-priority value;
        no-syslog;
        notification-targets address port port-number;
        service-port port-number;
        shared-key value;
        source-addresses [ addresses ];
    }
    duplicates-dropped-periodicity seconds;
    input-packet-rate-threshold rate;
```

```
    interfaces interface-name;
    max-duplicates number;
    pic-memory-threshold percentage percentage;
}
```

To specify the **capture-group**, assign it a unique ***client-name*** that associates the information with the requesting control sources.

Configuring the Content Destination

You must specify a destination for the packets that match DFC PIC filter criteria. To configure the content destination, include the **content-destination** statement at the **[edit services dynamic-flow-capture capture-group *client-name*]** hierarchy level:

```
content-destination identifier {
    address address;
    hard-limit bandwidth;
    hard-limit-target bandwidth;
    soft-limit bandwidth;
    soft-limit-clear bandwidth;
    ttl hops;
}
```

Assign the **content-destination** a unique ***identifier***. You must also specify its IP address and you can optionally include additional settings:

- **address**—The DFC PIC interface appends an IP header with this destination address on the matched packet (with its own IP header and contents intact) and sends it out to the content destination.
- **ttl**—The time-to-live (TTL) value for the IP-IP header. By default, the TTL value is 255. Its range is 0 through 255.
- Congestion thresholds—You can specify per-content destination bandwidth limits that control the amount of traffic produced by the DFC PIC during periods of congestion. The thresholds are arranged in two pairs: **hard-limit** and **hard-limit-target**, and **soft-limit** and **soft-limit-clear**. You can optionally include one or both of these paired settings. All four settings are 10–second average bandwidth values in bits per second. Typically **soft-limit-clear** < **soft-limit** < **hard-limit-target** < **hard-limit**. When the content bandwidth exceeds the **soft-limit** setting:

  1. A congestion notification message is sent to each control source of the criteria that point to this content destination
  2. If the control source is configured for **syslog**, a system log message is generated.
  3. A latch is set, indicating that the control sources have been notified. No additional notification messages are sent until the latch is cleared, when the bandwidth falls below the **soft-limit-clear** value.

When the bandwidth exceeds the **hard-limit** value:

1. Junos Capture Vision begins deleting criteria until the bandwidth falls below the **hard-limit-target** value.
2. For each criterion deleted, a CongestionDelete notification is sent to the control source for that criterion.
3. If the control source is configured for **syslog**, a log message is generated.

The application evaluates criteria for deletion using the following data:

- Priority—Lower priority criteria are purged first, after adjusting for control source minimum priority.
- Bandwidth—Higher bandwidth criteria are purged first.
- Timestamp—The more recent criteria are purged first.

Configuring the Control Source

You configure information about the control source, including allowed source addresses and destinations and authentication key values. To configure the control source information, include the **control-source** statement at the **[edit services dynamic-flow-capture capture-group *client-name*]** hierarchy level:

```
control-source identifier {
    allowed-destinations [ destination-identifiers ];
    minimum-priority value;
    no-syslog;
    notification-targets address port port-number;
    service-port port-number;
    shared-key value;
    source-addresses [ addresses ];
}
```

Assign the **control-source** statement a unique ***identifier***. You can also include values for the following statements:

- **allowed-destinations**—One or more content destination identifiers to which this control source can request that matched data be sent in its control protocol requests. If you do not specify any content destinations, all available destinations are allowed.
- **minimum-priority**—Value assigned to the control source that is added to the priority of the criteria in the DTCP ADD request to determine the total priority for the criteria. The lower the value, the higher the priority. By default, **minimum-priority** has a value of 0 and the allowed range is 0 through 254.
- **notification-targets**—One or more destinations to which the DFC PIC interface can log information about control protocol-related events and other events such as PIC bootup messages. You configure each **notification-target** entry with an IP **address** value and a User Datagram Protocol (UDP) **port** number.

- **service-port**—UDP port number to which the control protocol requests are directed. Control protocol requests that are not directed to this port are discarded by DFC PIC interfaces.
- **shared-key**—20-byte authentication key value shared between the control source and the DFC PIC monitoring platform.
- **source-addresses**—One or more allowed IP addresses from which the control source can send control protocol requests to the DFC PIC monitoring platform. These are /32 addresses.

Configuring the DFC PIC Interface

You specify the interface that interacts with the control sources configured in the same capture group. A Monitoring Services III PIC can belong to only one capture group, and you can configure only one PIC for each group.

To configure a DFC PIC interface, include the **interfaces** statement at the **[edit services dynamic-flow-capture capture-group *client-name*]** hierarchy level:

```
interfaces interface-name;
```

You specify DFC interfaces using the **dfc-** identifier at the **[edit interfaces]** hierarchy level. You must specify three logical units on each DFC PIC interface, numbered **0**, **1**, and **2**. You cannot configure any other logical interfaces.

- **unit 0** processes control protocol requests and responses.
- **unit 1** receives monitored data.
- **unit 2** transmits the matched packets to the destination address.

The following example shows the configuration necessary to set up a DFC PIC interface and intercept both IPv4 and IPv6 traffic:

```
[edit interfaces dfc-0/0/0]
unit 0 {
   family inet {
      filter {
         output high; #Firewall filter to route control packets
         # through 'network-control' forwarding class. Control packets
         # are loss sensitive.
      }
      address 10.1.0.0/32 { # DFC PIC address
         destination 10.36.100.1; # DFC PIC address used by
         # the control source to correspond with the
         # monitoring platform
      }
```

```
    }
  unit 1 { # receive data packets on this logical interface
    family inet; # receive IPv4 traffic for interception
    family inet6; # receive IPv6 traffic for interception
  }
  unit 2 { # send out copies of matched packets on this logical interface
    family inet;
  }
```

In addition, you must configure Junos Capture Vision to run on the DFC PIC in the correct chassis location. The following example shows this configuration at the **[edit chassis]** hierarchy level:

```
fpc 0 {
  pic 0 {
    monitoring-services application dynamic-flow-capture;
  }
}
```

For more information on configuring chassis properties, see the *Junos OS Administration Library*.

Configuring the Firewall Filter

You can specify the firewall filter to route control packets through the network control forwarding class. The control packets are loss sensitive. To configure the firewall filter, include the following statements at the **[edit]** hierarchy level:

```
firewall {
  family inet {
    filter high {
      term all {
        then forwarding-class network-control;
      }
    }
  }
}
```

Configuring System Logging

By default, control protocol activity is logged as a separate system log facility, **dfc**. To modify the filename or level at which control protocol activity is recorded, include the following statements at the **[edit syslog]** hierarchy level:

```
file dfc.log {
    dfc any;
}
```

To cancel logging, include the **no-syslog** statement at the **[edit services dynamic-flow-capture capture-group *client-name* control-source *identifier*]** hierarchy level:

```
no-syslog;
```

NOTE: Junos Capture Vision (**dfc-**) interface supports up to 10,000 filter criteria. When more than 10,000 filters are added to the interface, the filters are accepted, but system log messages are generated indicating that the filter is full.

Configuring Tracing Options for Junos Capture Vision Events

You can enable tracing options for Junos Capture Vision events by including the **traceoptions** statement at the **[edit services dynamic-flow-capture]** hierarchy level.

When you include the **traceoptions** configuration, you can also specify the trace file name, maximum number of trace files, the maximum size of trace files, and whether the trace file can be read by all users or not.

To enable tracing options for Junos Capture Vision events, include the following configuration at the **[edit services dynamic-flow-capture]** hierarchy level:

```
traceoptions{
    file filename <files number> <size size> <world-readable | non-world-readable>;
}
```

To disable tracing for Junos Capture Vision events, delete the **traceoptions** configuration from the **[edit services dynamic-flow-capture]** hierarchy level.

NOTE: In Junos OS releases earlier than 9.2R1, tracing of Junos Capture Vision was enabled by default, and the logs were saved to the**/var/log/dfcd** directory.

Configuring Thresholds

You can optionally specify threshold values for the following situations in which warning messages be recorded in the system log:

- Input packet rate to the DFC PIC interfaces
- Memory usage on the DFC PIC interfaces

To configure threshold values, include the **input-packet-rate-threshold** or **pic-memory-threshold** statements at the **[edit services dynamic-flow-capture capture-group *client-name*]** hierarchy level:

```
input-packet-rate-threshold rate;
pic-memory-threshold percentage percentage;
```

If these statements are not configured, no threshold messages are logged. The threshold settings are configured for the capture group as a whole.

The range of configurable values for the **input-packet-rate-threshold** statement is 0 through 1 Mpps. The PIC calibrates the value accordingly; the Monitoring Services III PIC caps the threshold value at 300 Kpps and the Multiservices 400 PIC uses the full configured value. The range of values for the **pic-memory-threshold** statement is 0 to 100 percent.

Limiting the Number of Duplicates of a Packet

You can optionally specify the maximum number of duplicate packets the DFC PIC is allowed to generate from a single input packet. This limitation is intended to reduce the load on the PIC when packets are sent to multiple destinations. When the maximum number is reached, the duplicates are sent to the destinations with the highest criteria class priority. Within classes of equal priority, criteria having earlier timestamps are selected first.

To configure this limitation, include the **max-duplicates** statement at the **[edit services dynamic-flow-capture capture-group *client-name*]** hierarchy level:

```
max-duplicates number;
```

You can also apply the limitation on a global basis for the DFC PIC by including the **g-max-duplicates** statement at the **[edit services dynamic-flow-capture]** hierarchy level:

```
g-max-duplicates number;
```

By default, the maximum number of duplicates is set to 3. The range of allowed values is 1 through 64. A setting for **max-duplicates** for an individual capture-group overrides the global setting.

In addition, you can specify the frequency with which the application sends notifications to the affected control sources that duplicates are being dropped because the threshold has been reached. You configure this setting at the same levels as the maximum duplicates settings, by including the **duplicates-dropped-periodicity** statement at the **[edit services dynamic-flow-capture capture-group *client-name*]** hierarchy level or the **g-duplicates-dropped-periodicity** statement at the **[edit services dynamic-flow-capture]** hierarchy level:

```
duplicates-dropped-periodicity seconds;
g-duplicates-dropped-periodicity seconds;
```

As with the **g-max-duplicates** statement, the **g-duplicates-dropped-periodicity** statement applies the setting globally for the application and is overridden by a setting applied at the capture-group level. By default, the frequency for sending notifications is 30 seconds.

RELATED DOCUMENTATION

Understanding Junos Capture Vision | 319

Example: Configuring Junos Capture Vision on M and T Series Routers | 329

Example: Configuring Junos Capture Vision on M and T Series Routers

The following example includes all parts of a complete Junos Capture Vision configuration.

Configure the Junos Capture Vision PIC interface:

```
[edit interfaces dfc-0/0/0]
unit 0 {
   family inet {
      filter {
         output high; #Firewall filter to route control packets
         # through 'network-control' forwarding class. Control packets
         # are loss sensitive.
      }
      address 10.1.0.0/32 { # DFC PIC address
         destination 10.36.100.1; # DFC PIC address used by
         # the control source to correspond with the
         # monitoring platform
      }
   }
unit 1 { # receive data packets on this logical interface
```

```
    family inet;
    family inet6;
}
unit 2 { # send out copies of matched packets on this logical interface
    family inet;
}
```

Configure the capture group:

```
services dynamic-flow-capture {
    capture-group g1 {
        interfaces dfc-0/0/0;
        input-packet-rate-threshold 90k;
        pic-memory-threshold percentage 80;
        control-source cs1 {
            source-addresses 10.36.41.1;
            service-port 2400;
            notification-targets {
                10.36.41.1 port 2100;
            }
            shared-key "$ABC123";
            allowed-destinations cd1;
        }
        content-destination cd1 {
            address 10.36.70.2;
            ttl 244;
        }
    }
}
```

Configur3 filter-based forwarding (FBF) to the Junos Capture Vision PIC interface, logical unit 1.

For more information about configuring passive monitoring interfaces, see “Enabling Passive Flow Monitoring on M Series, MX Series or T Series Routers” on page 203.

```
interfaces so-1/2/0 {
    encapsulation ppp;
    unit 0 {
        passive-monitor-mode;
        family inet {
            filter {
                input catch;
            }
```

```
        }
    }
}
```

Configure the firewall filter:

```
firewall {
    filter catch {
        interface-specific;
        term def {
            then {
                count counter;
                routing-instance fbf_inst;
            }
        }
    }
    family inet {
        filter high {
            term all {
                then forwarding-class network-control;
            }
        }
    }
}
```

Configure a forwarding routing instance. The next hop points specifically to the logical interface corresponding to **unit 1**, because only this particular logical unit is expected to relay monitored data to the Junos Capture Vision PIC.

```
routing-instances fbf_inst {
    instance-type forwarding;
    routing-options {
        static {
            route 0.0.0.0/0 next-hop dfc-0/0/0.1;
        }
    }
}
```

Configure routing table groups:

```
[edit]
routing-options {
    interface-routes {
```

```
        rib-group inet common;
    }
    rib-groups {
        common {
            import-rib [ inet.0 fbf_inst.inet.0 ];
        }
    }
    forwarding-table {
        export pplb;
    }
}
```

Configure interfaces to the control source and content destination:

```
interfaces fe-4/1/2 {
    description "to cs1 from dfc";
    unit 0 {
        family inet {
            address 10.36.41.2/30;
        }
    }
}
interfaces ge-7/0/0 {
    description "to cd1 from dfc";
    unit 0 {
        family inet {
            address 10.36.70.1/30;
        }
    }
}
```

RELATED DOCUMENTATION

Understanding Junos Capture Vision | 319

Configuring Junos Capture Vision | 321

Monitoring a Capture Group Using SNMP or Show Services Commands

In Junos OS Release 7.5 and later, the Dynamic Flow Capture MIB provides a way to monitor dynamic flow capture information by using Simple Network Management Protocol (SNMP). The MIB provides the same information that you can view with the **show services dynamic-flow-capture content-destination**, **show services dynamic-flow-capture control-source**, and **show services dynamic-flow-capture statistics** commands. For more information, see the *Junos Network Management Configuration Guide*.

CHAPTER 7

Detecting Threats and Intercepting Flows Using Junos Packet Vision

IN THIS CHAPTER

Understanding Junos Packet Vision

Junos Capture Vision (previously known as dynamic flow capture) enables you to capture packet flows on the basis of dynamic filtering criteria, using Dynamic Tasking Control Protocol (DTCP) requests. Junos Packet Vision is a Junos OS application that performs lawful intercept of packet flows, using Dynamic Tasking Control Protocol (DTCP). The application extends the use of DTCP to intercept IPv4 and IPv6 packets in an active monitoring router and send a copy of packets that match filter criteria to one or more content destinations. Junos Packet Vision was previously know as flow-tap application.

Junos Packet Vision data can be used in the following applications:

- Flexible trend analysis for detection of new security threats
- Lawful intercept

Junos Packet Vision is supported on M Series and T Series routers, except M160 and TX Matrix routers. Junos Packet Vision filters are applied on all IPv4 traffic and do not add any perceptible delay in the forwarding path. Junos Packet Vision filters can also be applied on IPv6 traffic. For security, filters installed by one client are not visible to others and the CLI configuration does not reveal the identity of the monitored target. A lighter version of the application is supported on MX Series routers only.

RELATED DOCUMENTATION

Junos Packet Vision Architecture

Configuring Junos Packet Vision on MX, M and T Series Routers

IN THIS SECTION

This topic explains Junos Packet Vision (previously known as Flow-Tap) configuration, and contains the following sections:

Configuring the Junos Packet Vision Interface

To configure an adaptive services interface for flow-tap service, include the **interface** statement at the **[edit services flow-tap]** hierarchy level:

```
interface sp-fpc/pic/port.unit-number;
```

You can assign any Adaptive Services or Multiservices PIC in the active monitoring router for Junos Packet Vision, and use any logical unit on the PIC.

You can specify the type of traffic for which you want to apply the Junos Packet Vision service by including the **family inet | inet6** statement. If the **family** statement is not included, the Junos Packet Vision service is, by default, applied to the IPv4 traffic. To apply Junos Packet Vision service to IPv6 traffic, you must include the **family inet6** statement in the configuration. To enable the Junos Packet Vision service for IPv4 and IPv6 traffic, you must explicitly configure the **family** statement for both **inet** and **inet6** families.

NOTE: You cannot configure Junos Capture Vision (previously known as dynamic flow capture) and Junos Packet Vision services on the same router simultaneously.

You must also configure the logical interface at the **[edit interfaces]** hierarchy level:

```
interface sp-fpc/pic/port {
    unit logical-unit-number {
        family inet;
        family inet6;
    }
}
```

> **NOTE:** If you do not include the **family inet6** statement in the configuration, IPv6 flows are not intercepted. Note that the Flow-Tap solution did not support IPv6.

Strengthening Junos Packet Vision Security

You can add an extra level of security to Dynamic Tasking Control Protocol (DTCP) transactions between the mediation device and the router by enabling DTCP sessions on top of the SSH layer. To configure SSH settings, include the **flow-tap-dtcp** statement at the **[edit system services]** hierarchy level:

```
flow-tap-dtcp {
    ssh {
        connection-limit value;
        rate-limit value;
    }
}
```

To configure client permissions for viewing and modifying Junos Packet Vision configurations and for receiving tapped traffic, include the **permissions** statement at the **[edit system login class *class-name*]** hierarchy level:

```
permissions [permissions];
```

The permissions needed to use Junos Packet Vision features are as follows:

- **flow-tap**—Can view Junos Packet Vision configuration
- **flow-tap-control**—Can modify Junos Packet Vision configuration
- **flow-tap-operation**—Can tap flows

You can also specify user permissions on a RADIUS server, for example:

```
Bob Auth-Type := Local, User-Password = = “abc123”
Juniper-User-Permissions = “flow-tap-operation”
```

Starting in Junos OS Release 16.2, MX Series routers can process mediation device DTCP ADD requests that contain up to 15 source-destination port pairs. Multiple source-destination port pairs must be separated by commas. For example:

```
ADD DTCP/0.7
Csource-ID: ftap
Cdest-ID: cd2
Source-Port: 2000,8001,4000,5000,6000,6001,6002
Dest-Port: 2000,9001,4000,5000,6000,9000
```

For details on **[edit system]** and RADIUS configuration, see the *Junos OS Administration Library*.

Restrictions on Junos Packet Vision Services

The following restrictions apply to Junos Packet Vision services:

- You cannot configure Junos Capture Vision and Junos Packet Vision features on the same router simultaneously.
- On routers that support LMNR-based FPCs, you cannot configure the Junos Packet Vision for IPv6 along with port mirroring or sampling of IPv6 traffic. This restriction applies even if the router does not have any LMNR-based FPC installed in it. However, there is no restriction on configuring Junos Packet Vision on routers that are configured for port mirroring or sampling of IPv4 traffic.
- Junos Packet Vision does not support interception of MPLS and virtual private LAN service (VPLS).
- Junos Packet Vision cannot intercept Address Resolution Protocol (ARP) and other Layer 2 exceptions.
- IPv4 and IPv6 intercept filters can coexist on a system, subject to a combined maximum of 100 filters.
- When Junos Capture Vision process or the Adaptive Services or Multiservices PIC configured for Junos Packet Vision restarts, all filters are deleted and the mediation devices are disconnected.
- Only the first fragment of an IPv4 fragmented packet stream is sent to the content destination.
- Port mirroring might not work in conjunction with Junos Packet Vision.
- Running the Junos Packet Vision over an IPsec tunnel on the same router can cause packet loops and is not supported.
- M10i routers do not support the standard Junos Packet Vision, but do support FlowTapLite (see “Configuring FlowTapLite on MX Series Routers and M320 Routers with FPCs” on page 357). Junos Packet Vision and FlowTapLite cannot be configured simultaneously on the same chassis.

- PIC-based flow-tap is not supported on M7i and M10i routers equipped with an Enhanced Compact Forwarding Engine Board (CFEB-E).
- You cannot configure Junos Packet Vision on channelized interfaces.

Release History Table

Release	Description
16.2	Starting in Junos OS Release 16.2, MX Series routers can process mediation device DTCP ADD requests that contain up to 15 source-destination port pairs. Multiple source-destination port pairs must be separated by commas.

RELATED DOCUMENTATION

Configuring FlowTapLite on MX Series Routers and M320 Routers with FPCs | 357

Examples: Configuring Junos Packet Vision on M, T, and MX Series Routers

The following example shows all parts of a complete Junos Packet Vision configuration with IPv4 and IPv6 flow intercepts:

NOTE: The following example applies only to M Series and T Series routers, except M160 and TX Matrix routers. For MX Series routers, because the flow-tap application resides in the Packet Forwarding Engine rather than a service PIC or Dense Port Concentrator (DPC), the Packet Forwarding Engine must send the packet to a tunnel logical (vt-) interface to encapsulate the intercepted packet. In such a scenario, you need to allocate a tunnel interface and assign it to the dynamic flow capture process for FlowTapLite to use.

```
services {
    flow-tap {
        interface sp-1/2/0.100;
    }
}
interfaces {
    sp-1/2/0 {
        unit 100 {
```

```
            family inet;
            family inet6;
        }
    }
}
system {
    services {
        flow-tap-dtcp {
            ssh {
                connection-limit 5;
                rate-limit 5;
            }
        }
    }
    login {
        class ft-class {
            permissions flow-tap-operation;
        }
        user ft-user1 {
            class ft-class;
            authentication {
                encrypted-password “xxxx”;
            }
        }
    }
}
```

The following example shows a FlowTapLite configuration that intercepts IPv4 and IPv6 flows:

```
system {
    login {
        class flowtap {
            permissions flow-tap-operation;
        }
        user ftap {
            uid 2000;
            class flowtap;
            authentication {
                encrypted-password "$ABC123"; ## SECRET-DATA
            }
        }
    }
    services {
        flow-tap-dtcp {
```

```
            ssh;
        }
    }
}
    chassis {
        fpc 0 {
            pic 0 {
                tunnel-services {
                    bandwidth 10g;
                }
            }
        }
    }
    interfaces {
        vt-0/0/0 {
            unit 0 {
                family inet;
                family inet6;
            }
        }
    }
    services {
        flow-tap {
            tunnel-interface vt-0/0/0.0;
        }
    }
```

RELATED DOCUMENTATION

- Understanding Junos Packet Vision | 334
- Configuring Junos Packet Vision on MX, M and T Series Routers | 335
- Configuring FlowTapLite on MX Series Routers and M320 Routers with FPCs | 357

Sending Packets to a Mediation Device on MX, M and T Series Routers

Dynamic flow capture enables you to capture passively monitored packet flows on the basis of dynamic filtering criteria, using Dynamic Tasking Control Protocol (DTCP) requests. The flow-tap application extends the use of DTCP to intercept IPv4 packets in an active flow monitoring station and send a copy of packets that match filter criteria to one or more content destinations. Flow-tap data can be used for

lawful intercept purposes and provides flexible trend analysis for detection of new security threats. The flow-tap application is supported on M Series and T Series routers, except M160 routers and TX Matrix platforms.

> **NOTE:** . For information about DTCP, see Internet draft draft-cavuto-dtcp-01.txt at **http://www.ietf.org/internet-drafts**.

For detailed information about the flow-tap application, see the following sections:

- Understanding Flow-Tap Architecture on page 351
- Configuring a Flow-Tap Interface on MX, M and T Series Routers on page 353
- Configuring Flow-Tap Security Properties on MX, M and T Series Routers on page 354
- Flow-Tap Application Restrictions on page 355
- Example: Flow-Tap Configuration on T and M Series Routers on page 356

Example: Configuring IPv6 Support for FlowTapLite on an M120 Router With Enhanced III FPCs

IN THIS SECTION

This example describes how to configure IPv6 support for FlowTapLite on an M120 router with Enhanced III FPCs. The configuration of FlowTapLite is similar on an M320 router and an MX Series router with Enhanced III FPCs. However, because the MX Series routers do not support Tunnel Services PICs, you configure a DPC and the corresponding Packet Forwarding Engine to use tunneling services at the **[edit chassis]** hierarchy level.

With Junos OS Release 10.1, the FlowTapLite service supports lawful interception of IPv6 packets; previously only interception of IPv4 packets was supported. The intercepted packets are sent to a content destination, while the flow of original packets to the actual destination is unaffected.

A mediation device installs dynamic filters on the router (or server) by sending DTCP requests. These filters include the quintuple information (source address, destination address, source port, destination port, and protocol) about the intercepted flows and the details (IP addresses and port information) of the content destination.

Below is an example of such a filter:

```
ADD DTCP/0.8
Csource-ID: ftap
Cdest-ID: cd1
Source-Address: 2001:0DB8:ABCD:EF12:3456:78AB:ABC8:1235/112
Dest-Address: affe::1:1
Source-Port: 1234
Dest-Port: 2345
Protocol: *
Priority: 2
X-JTap-Input-Interface: ge-2/0/1
X-JTap-Cdest-Dest-Address: 192.0.2.5
X-JTap-Cdest-Dest-Port: 2300
X-JTap-Cdest-Source-Address: 198.51.100.9
X-JTap-Cdest-Source-Port: 65535
X-JTap-Cdest-TTL: 255
X-JTap-IP-Version: ipv6
Flags: STATIC
```

Following are descriptions of the parameters in the dynamic filter:

- **Csource-ID**—The username configured in the router at the [**edit system login user**] hierarchy level.
- **Cdest-ID**—The content destination identifier.
- **Source-Address**, **Dest-Address Source-Port**, **Dest-Port**, **Protocol**—Parameters that determine which packet flows need to be intercepted.
- **X-JTap-Input-Interface**—The interface through which the actual flows are coming into the router. Depending on the type of filters installed, the value in this field can include the following: **X-JTap-Output-Interface** to install output interface filters; **X-JTap-VRF-NAME** to install VRF filters; and to install global filters, no parameters are specified.

- **X-JTap-Cdest-Dest**—All parameters that start with this string specify different parameters associated with the content destination.
- **X-JTap-IP-Version**–Differentiates between IPv6 and IPv4 filters.

From the Packet Forwarding Engine console, you can verify that the filters are installed and working correctly.

This example describes how to configure IPv6 support for FlowTapLite on an M120 router:

Requirements

This example uses the following hardware and software components:

- Junos OS Release 10.1 or later
- M120 router with a tunnel (**vt**) interface

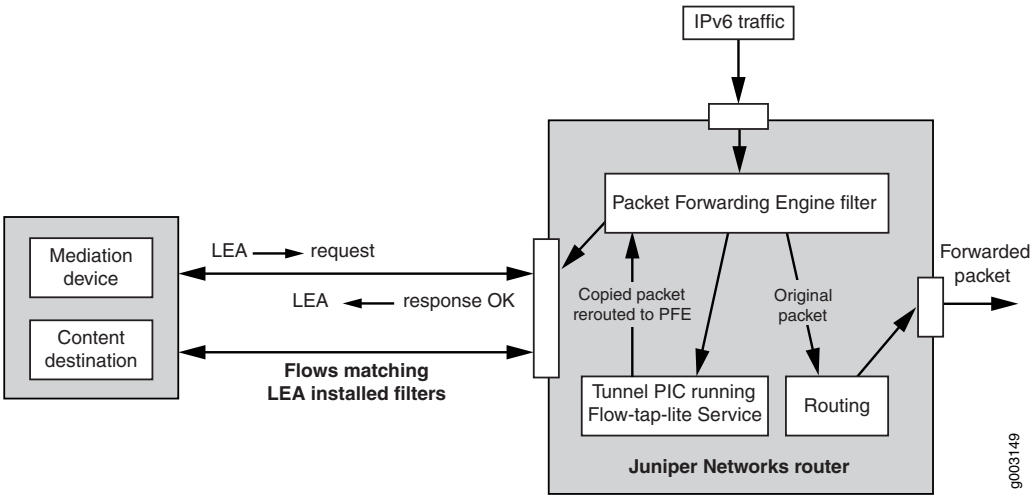
Before you configure IPv6 FlowTapLite on your router, be sure you have:

- A tunnel PIC that is up
- A connection from the router to the mediation device and the content destination
- Traffic flow to and from the router

Overview and Topology

Figure 31 on page 343 shows the FlowTapLite configuration for one M120 router to lawfully intercept packets.

Figure 31: FlowTapLite Topology



In this example, the IPv6 packets enter the Packet Forwarding Engine and, depending on the filters installed, a new flow is created for the intercepted packets while the original packets are forwarded normally. The new flow is rerouted through the tunnel PIC back to the Packet Forwarding Engine for a route lookup, and then on to the content destination.

Configuration

IN THIS SECTION

To configure IPv6 FlowTapLite on an M120 router, perform these tasks:

CLI Quick Configuration

To quickly configure IPv6 FlowTapLite, copy the following commands and paste them into the CLI:

```
set system login class flowtap permissions flow-tap-operation
set system login user ftap uid 2000
set system login user ftap class flowtap
set system login user ftap authentication encrypted-password "xxxxxx"
set system services flow-tap-dtcp ssh
set interfaces vt-4/0/0 unit 0 family inet
set interfaces vt-4/0/0 unit 0 family inet6
set services flow-tap tunnel-interface vt-4/0/0.0
```

Configuring User Credentials

Step-by-Step Procedure

The username and password configured here are used by the mediation device when connecting and sending out DTCP requests.

1. Define a login class called **flowtap**:

```
[edit system]
user@router# set login class flowtap permissions flow-tap-operation
```

2. For the meditation device, configure a user called **ftap** with a unique identifier (UID):

```
[edit system]
user@router# set login user ftap uid 2000
```

3. Apply the **flowtap** class to the **ftap** user:

```
[edit system]
user@router# set login user ftap class flowtap
```

4. Configure the password used by the mediation device:

```
[edit system]
user@router# set login user ftap authentication encrypted-password xxxxxx
```

5. Commit the configuration:

```
[edit system]
user@router# commit
```

Configuring the Tunnel Interface for FlowTapLite

Step-by-Step Procedure

You can add an extra level of security to DTCP transactions between the mediation device and the router by enabling DTCP sessions on top of the SSH layer.

1. Configure SSH from the **[edit system]** hierarchy level:

```
[edit system]
user@router# set services flow-tap-dtcp ssh
```

2. Commit the configuration:

```
[edit system]
user@router# commit
```

Configuring the Logical Tunnel Interface

Step-by-Step Procedure

1. Configure the logical interface and assign it to the dynamic flow control process (dfcd) at the **[edit interfaces]** hierarchy level:

```
[edit interfaces]
user@router# set vt-4/0/0 unit 0 family inet
```

2. Include the mandatory **inet6** statement:

```
[edit interfaces]
user@router# set vt-4/0/0 unit 0 family inet6
```

3. Commit the configuration:

```
[edit interfaces]
user@router# commit
```

Configuring FlowTapLite

Step-by-Step Procedure

1. Include the **flow-tap** statement and the tunnel interface at the **[edit services]** hierarchy level:

```
[edit services]
user@router# set flow-tap tunnel-interface vt-4/0/0.0
```

2. Commit the configuration:

```
[edit services]
user@router# commit
```

Results

Check the results of the configuration:

```
[edit]
user@router# show
system {
    [...Output Truncated...]
    login {
        class flowtap {
            permissions flow-tap-operation;
```

```
        }
        user ftap {
            uid 2000;
            class flowtap;
            authentication {
                encrypted-password "xxxxxx"; ## SECRET-DATA
            }
        }
    }
    services {
        telnet;
        flow-tap-dtcp {
            ssh;
        }
    }
}
interfaces {
    vt-4/0/0 {
        unit 0 {
            family inet;
            family inet6;
        }
    }
}
[...Output Truncated...]
services {
    flow-tap {
        tunnel-interface vt-4/0/0.0;
    }
}
```

Verification

IN THIS SECTION

To confirm that the configuration is working properly, perform the following tasks:

Verifying That the Router Received the Filter Request

Purpose

After the mediation device sends the filters to the router, the mediation device must receive a message from the router confirming that the router has received the filter request.

Action

Check that the mediation device has received a message similar to the one below:

```
DTCP/0.8 200 OK
SEQ: 1
CRITERIA-ID: 1
TIMESTAMP: 2009-09-29 06:12:05.725
AUTHENTICATION-INFO: 55f9dc3debd3c7356951410f165f2a9cc5606063
```

Meaning

The message above is an example of a successfully received filter request.

Checking That Filters Are Installed and Working on the Router

Action

Use the **show filter** and the **show filter index** commands to check that filters are installed:

```
ADPC2(diving vty)# show filter
Program Filters:
---------------
   Index     Dir     Cnt    Text     Bss  Name
--------  ------  ------  ------  ------  --------
       1     104       0      20      20  __default_bpdu_filter__
   17000      52       0       4       4  __default_arp_policer__
   57007     104     144      16      16  __flowtap_inet__
   65280      52       0       4       4  __auto_policer_template__
   65281     104       0      16      16  __auto_policer_template_1__
   65282     156       0      32      32  __auto_policer_template_2__
   65283     208       0      48      48  __auto_policer_template_3__
   65284     260       0      64      64  __auto_policer_template_4__
   65285     312       0      80      80  __auto_policer_template_5__
   65286     364       0      96      96  __auto_policer_template_6__
   65287     416       0     112     112  __auto_policer_template_7__
   65288     468       0     128     128  __auto_policer_template_8__
37748736   156   144   80   80 __ftaplite_filter__ifl__70__out__ipv6_
```

```
37748737   156   144    80    80 __ftaplite_filter__vrf__4__in__ipv6_
37748738   156   144    80    80 __ftaplite_filter__ifl__71__in__ipv6_
37748739   156   144    80    80 __ftaplite_filter__vrf__0__in__ipv6_

ADPC2(diving vty)# show filter index 37748738 counters
Filter Counters/Policers:
   Index               Packets                 Bytes  Name
--------  -------------------  -------------------  --------
37748738               8851815            601923420
__ftaplite_term_ftap_3___counter
```

Meaning

The last four filters in the output for the **show filter** command above are the filters installed on the Packet Forwarding Engine. The **show filter index** command shows a non-zero packet count, indicating that the packets are hitting the filter.

Sending a List Request

Purpose

To verify that the correct filters are installed in the Packet Forwarding Engine.

Action

Use client software to send a list request to the Packet Forwarding Engine. In your list request, you can include the following three parameters individually or together: **CSource-Id**, **CDest-ID**, and **Criteria-ID**. With all requests, you must include the **CSource-Id**. Below is an example of a list request using the **CSource-Id**:

```
LIST DTCP/0.8
Csource-ID: ftap1
Flags: Both
```

Below is an example of a response:

```
DTCP/0.8 200 OK
SEQ: 51
TIMESTAMP: 2009-10-04 07:56:43.003
CRITERIA-ID: 1
CSOURCE-ID: ftap1
CDEST-ID: cd1
CSOURCE-ADDRESS: 10.209.152.15
FLAGS: Static
```

```
AVERAGE-BANDWIDTH: 0
MATCHING-PACKETS: 0
MATCHING-BYTES: 0
NUM-REFRESH: 0
LAST-REFRESH: 2009-10-04 07:54:30.870
X-JTAP-INPUT-INTERFACE: ge-2/1/1.0,ge-2/1/1.1,ge-2/1/1.2
SOURCE-ADDRESS: 203.0.113.1
DEST-ADDRESS: 192.168.0.1/32
SOURCE-PORT: 1000
DEST-PORT: 2000
PROTOCOL: 17
X-JTAP-CDEST-DEST-ADDRESS: 192.168.99.81
X-JTAP-CDEST-DEST-PORT: 8001
X-JTAP-CDEST-SOURCE-ADDRESS: 192.168.208.9
X-JTAP-CDEST-SOURCE-PORT: 34675
X-JTAP-CDEST-TTL: 64
CRITERIA-NUM: 1
CRITERIA-COUNT: 1
AUTHENTICATION-INFO: 0f49ff600a3d8d7d312c5031f74cc17540bc9200
```

You can also delete the request. Below is an example of a delete request:

```
DELETE DTCP/0.8
Csource-ID: ftap
CDest-ID: cd1
Flags: STATIC
```

RELATED DOCUMENTATION

Configuring FlowTapLite on MX Series Routers and M320 Routers with FPCs | **357**

flow-tap | **975**

Tunnel Interface Configuration on MX Series Routers Overview

CHAPTER 8

Using Flow-Tap to Monitor Packet Flow

IN THIS CHAPTER

Understanding Flow-Tap Architecture

The flow-tap architecture consists of one or more *mediation devices* that send requests to a Juniper Networks router to monitor incoming data. Any packets that match specific filter criteria are forwarded to a set of one or more *content destinations*:

- Mediation device—A client that monitors electronic data or voice transfer over the network. The mediation device sends filter requests to the Juniper Networks router using the DTCP. The clients are not identified for security reasons, but have permissions defined by a set of special login classes.
- Monitoring platform—A Juniper Networks M Series or T Series router containing one or more Adaptive Services (AS) PICs, which are configured to support the flow-tap application. The monitoring platform processes the requests from the mediation devices, applies the dynamic filters, monitors incoming data flows, and sends the matched packets to the appropriate content destinations.
- Content destination—Recipient of the matched packets from the monitoring platform. Typically the matched packets are sent using an IP Security (IPSec) tunnel from the monitoring platform to another router connected to the content destination. The content destination and the mediation device can be physically located on the same host.
- Dynamic filters—The Packet Forwarding Engine automatically generates a firewall filter that is applied to all IPv4 routing instances. Each term in the filter includes a **flow-tap** action that is similar to the existing **sample** or **port-mirroring** actions. As long as one of the filter terms matches an incoming packet, the router copies the packet and forwards it to the AS PIC that is configured for flow-tap service. The AS PIC runs the packet through the client filters and sends a copy to each matching content destination.

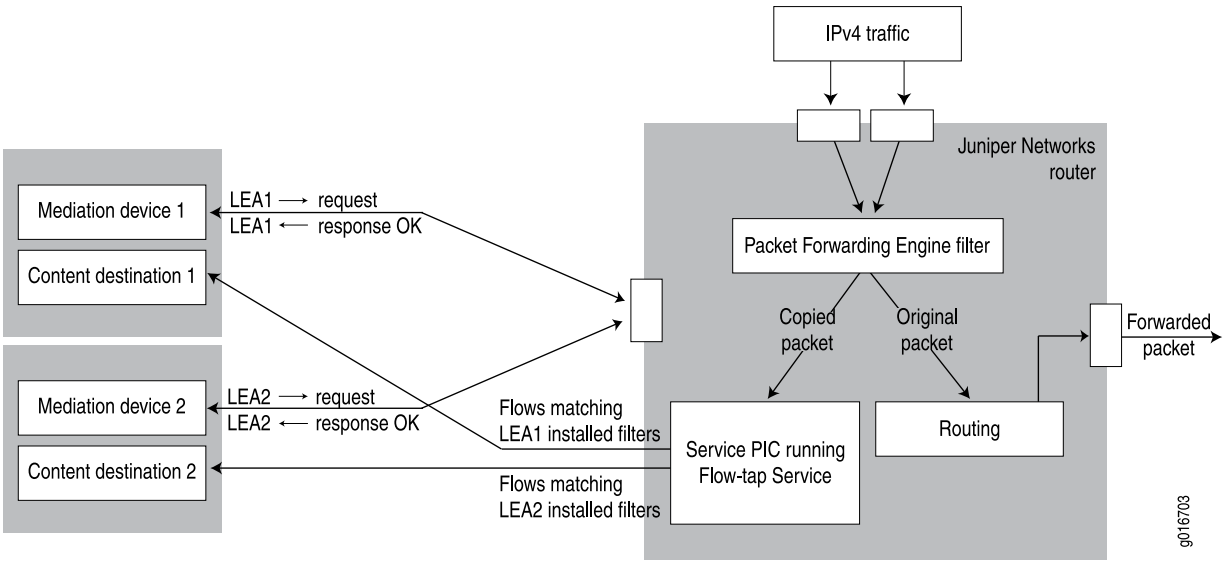
For security, filters installed by one client are not visible to others and the CLI configuration does not reveal the identity of the monitored target.

Following is a sample filter configuration; note that it is dynamically generated by the router (no user configuration is required):

```
filter combined_LEA_filter {
    term LEA1_filter {
        from {
            source-address 192.0.2.;
            destination-address 198.51.100.6;
        }
        then {
            flow-tap;
        }
    }
    term LEA2_filter {
        from {
            source-address 10.1.1.1;
            source-port 23;
        }
        then {
            flow-tap;
        }
    }
}
```

Figure 32 on page 353 shows a sample topology that uses two mediation devices and two content destinations.

Figure 32: Flow-Tap Topology Diagram



RELATED DOCUMENTATION

Configuring a Flow-Tap Interface on MX, M and T Series Routers

To configure an AS PIC interface for the flow-tap service, include the **interface** statement at the [**edit services flow-tap**] hierarchy level:

```
interface sp-fpc/pic/port.unit-number;
```

You can assign any AS PIC in the active monitoring station for flow-tap service, and use any logical unit on the PIC.

NOTE: You cannot configure dynamic flow capture and flow-tap features on the same router simultaneously.

You must also configure the logical interface at the **[edit interfaces]** hierarchy level:

```
interface sp-fpc/pic/port {
    unit logical-unit-number {
        family inet;
    }
}
```

RELATED DOCUMENTATION

- Understanding Flow-Tap Architecture | 351
- Configuring Flow-Tap Security Properties on MX, M and T Series Routers | 354
- Flow-Tap Application Restrictions | 355
- Example: Flow-Tap Configuration on T and M Series Routers | 356

Configuring Flow-Tap Security Properties on MX, M and T Series Routers

You can add an extra level of security to DTCP transactions between the mediation device and the router by enabling DTCP sessions on top of the SSH layer. To configure, include the **flow-tap-dtcp** statement at the [**edit system services**] hierarchy level:

```
flow-tap-dtcp {
    ssh {
        connection-limit value;
        rate-limit value;
    }
}
```

To configure client permissions for viewing and modifying flow-tap configurations and for receiving tapped traffic, include the **permissions** statement at the **[edit system login class *class-name*]** hierarchy level:

```
permissions [ permissions ];
```

The permissions needed to use flow-tap features are as follows:

- **flow-tap**—Can view flow-tap configuration.
- **flow-tap-control**—Can modify flow-tap configuration.
- **flow-tap-operation**—Can tap flows.

You can also specify user permissions on a RADIUS server, for example:

```
Bob Auth-Type := Local, User-Password = = “abc123”
Juniper-User-Permissions = “flow-tap-operation”
```

For details on **[edit system]** and RADIUS configuration, see the *Junos System Basics Configuration Guide*.

RELATED DOCUMENTATION

- Understanding Flow-Tap Architecture | **351**
- Configuring a Flow-Tap Interface on MX, M and T Series Routers | **353**
- Flow-Tap Application Restrictions | **355**
- Example: Flow-Tap Configuration on T and M Series Routers | **356**

Flow-Tap Application Restrictions

The following restrictions apply to flow-tap services:

- You cannot configure dynamic flow capture and flow-tap services on the same router simultaneously.
- When the dynamic flow capture process or an AS PIC configured for flow-tap processing restarts, all filters are deleted and the mediation devices are disconnected.
- Only the first fragment of an IPv4 fragmented packet stream is sent to the content destination.
- If the flow-tap application is configured, you cannot configure the filter action **then syslog** for any firewall filter running on the same platform.
- Running the flow-tap application over an IPsec tunnel on the same router can cause packet loops and is not supported.
- The flow-tap service **[edit services flow-tap]** on tunnel interfaces on MX Series routers (FlowTapLite) and the RADIUS flow-tap service **[edit services radius-flow-tap]** cannot run simultaneously on the router. Consequently, you cannot run both FlowTapLite and subscriber secure policy mirroring at the same time on the same router in the earlier releases. However, starting in Junos OS Release 17.3R1, FlowTapLite and subscriber secure policy mirroring are supported to run concurrently on the same MX Series router.

Release History Table

Release	Description
17.3R1	However, starting in Junos OS Release 17.3R1, FlowTapLite and subscriber secure policy mirroring are supported to run concurrently on the same MX Series router.

RELATED DOCUMENTATION

Example: Flow-Tap Configuration on T and M Series Routers

The following example shows all the parts of a complete flow-tap configuration.

> **NOTE:** The following example applies only to M Series and T Series routers, except M160 and TX Matrix routers. For MX Series routers, because the flow-tap application resides in the Packet Forwarding Engine rather than a service PIC or Dense Port Concentrator (DPC), the Packet Forwarding Engine must send the packet to a tunnel logical (vt-) interface to encapsulate the intercepted packet. In such a scenario, you need to allocate a tunnel interface and assign it to the dynamic flow capture process for FlowTapLite to use.

```
services {
    flow-tap {
        interface sp-1/2/0.100;
    }
}
interfaces {
    sp-1/2/0 {
        unit 100 {
            family inet;
        }
    }
}
system {
    services {
        flow-tap-dtcp {
            ssh {
                connection-limit 5;
                rate-limit 5;
            }
        }
```

```
    }
    login {
        class ft-class {
            permissions flow-tap-operation;
        }
        user ft-user1 {
            class ft-class;
            authentication {
                encrypted-password “xxxx”;
            }
        }
    }
}
```

RELATED DOCUMENTATION

- Understanding Flow-Tap Architecture | 351
- Configuring a Flow-Tap Interface on MX, M and T Series Routers | 353
- Configuring Flow-Tap Security Properties on MX, M and T Series Routers | 354
- Flow-Tap Application Restrictions | 355

Configuring FlowTapLite on MX Series Routers and M320 Routers with FPCs

A lighter version of the flow-tap application is available on MX Series routers and also on M320 routers with Enhanced III Flexible PIC Concentrators (FPCs). All of the functionality resides in the Packet Forwarding Engine rather than in a service PIC or Dense Port Concentrator (DPC).

Starting in Junos OS Release 17.2R1, FlowTapLite supports the sampling of circuit cross connect (CCC) traffic.

Starting in Junos OS Release 19.3R1, you can configure FlowTapLite on MX240, MX480, and MX960 routers with an MPC10E line card.

NOTE: On M320 routers only, if the replacement of FPCs results in a mode change, you must restart the dynamic flow capture process manually by disabling and then re-enabling the CLI configuration.

FlowTapLite uses the same DTCP-SSH architecture to install the Dynamic Tasking Control Protocol (DTCP) filters and authenticate the users as the original flow-tap application and supports up to 3000 filters per chassis.

NOTE: The original flow-tap application and FlowTapLite cannot be used at the same time.

To configure FlowTapLite, include the **flow-tap** statement at the **[edit services]** hierarchy level:

```
flow-tap {
    tunnel-interface interface-name;
}
```

If you do not specify a family, FlowTapLite is applied only to IPv4 traffic. Starting in Junos OS release 17.2R1, FlowTapLite can be applied to circuit cross connect traffic (ccc).

For the Packet Forwarding Engine to encapsulate the intercepted packet, it must send the packet to a tunnel logical (**vt-**) interface. You need to allocate a tunnel interface and assign it to the dynamic flow capture process for FlowTapLite to use. To create the tunnel interface, include the following configuration:

```
chassis {
    fpc number {
        pic number {
            tunnel-services {
                bandwidth (1g | 10g);
            }
        }
    }
}
```

NOTE: Currently FlowTapLite supports only one tunnel interface per instance.

For more information about this configuration, see the *Junos OS Administration Library*.

To configure the logical interfaces and assign them to the dynamic flow capture process, include the following configuration:

```
interfaces {
    vt-fpc/pic/port {
        unit 0 {
```

```
            family inet;
            family inet6;
        }
    }
}
```

NOTE: If a service PIC or DPC is available, you can use its tunnel interface for the same purpose.

NOTE: If you do not include the **family intet6** statement in the configuration, IPv6 flows are not intercepted.

NOTE: With FlowTapLite configured and traceoptions enabled, if you add more than two content destinations by including the X-JTAP- CDEST-DEST-ADDRESS line in the Dynamic Tasking Control Protocol (DTCP) parameter file and initiate a DTCP session by sending a DTCP ADD message, a **400 BAD request** message is received. Although you can specify more than two content destinations in the DTCP file that is sent from the mediation device, this error message occurs when the DTCP ADD message is sent. This behavior is expected with more than two content destinations. You must specify only two content destinations per DTCP ADD message.

The FlowTapLite service **[edit services flow-tap]** and the RADIUS flow-tap service **[edit services radius-flow-tap]** cannot run simultaneously on the router. Consequently, you cannot run both FlowTapLite and subscriber secure policy mirroring at the same time on the same router. Starting in Junos OS Release 17.3R1, FlowTapLite and subscriber secure policy mirroring are supported to run concurrently on the same MX Series router.

Release History Table

Release	Description
19.3R1	Starting in Junos OS Release 19.3R1, you can configure FlowTapLite on MX240, MX480, and MX960 routers with an MPC10E line card.
17.3R1	Starting in Junos OS Release 17.3R1, FlowTapLite and subscriber secure policy mirroring are supported to run concurrently on the same MX Series router.
17.2R1	Starting in Junos OS Release 17.2R1, FlowTapLite supports the sampling of circuit cross connect (CCC) traffic.
17.2R1	Starting in Junos OS release 17.2R1, FlowTapLite can be applied to circuit cross connect traffic (ccc).

RELATED DOCUMENTATION

Understanding Junos Packet Vision | 334

Configuring Junos Packet Vision on MX, M and T Series Routers | 335

Examples: Configuring Junos Packet Vision on M, T, and MX Series Routers | 338

Subscriber Secure Policy Overview

3

PART

Inline Monitoring Services

CHAPTER 9

Configuring Inline Monitoring Services

IN THIS CHAPTER

Inline Monitoring Services Configuration

IN THIS SECTION

Understanding Inline Monitoring Services

IN THIS SECTION

Benefits of Inline Monitoring Services

Flexible—Inline monitoring services allow different inline-monitoring instances to be mapped to different firewall filter terms, unlike in traditional sampling technologies, where all the instances are mapped to the Flexible PIC Concentrator (FPC). This provides you with the flexibility of sampling different streams of traffic at different rates on a single interface.

Packet format agnostic—Traditional flow collection technologies rely on packet parsing and aggregation by the network element. With inline monitoring services, the packet header is exported to the collector for further processing, but without aggregation. Thereby, you have the benefit of using arbitrary packet fields to process the monitored packets at the collector.

Inline Monitoring Services Feature Overview

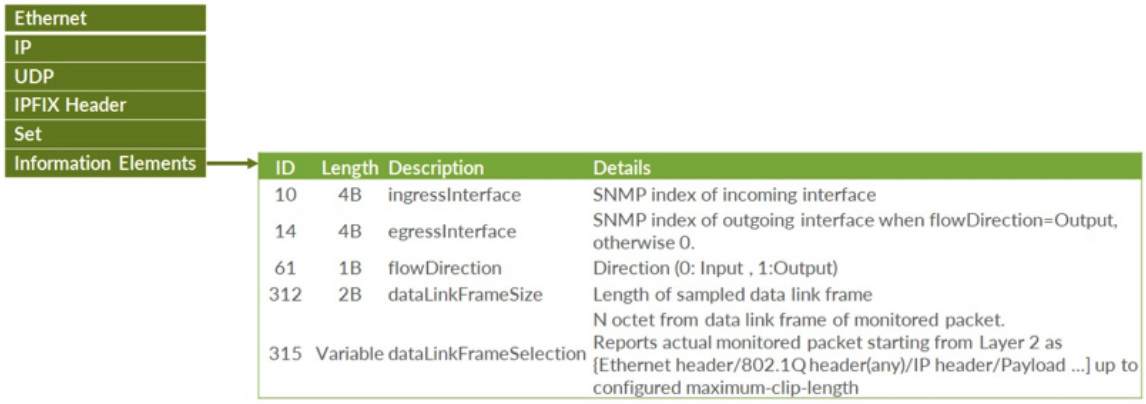
Service providers and content providers typically require visibility into traffic flows in order to evaluate peering agreements, detect traffic anomalies and policy violations, and monitor network performance. To meet these requirements, you would traditionally export aggregate flow statistics information using Netflow, JFlow, or IPFIX variants.

As an alternative approach, you can have the packet content sampled, add metadata information, and export the monitored packets to an collector. The inline monitoring services enables you to do this on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

With inline monitoring services, you can monitor every IPv4 and IPv6 packet on both ingress and egress directions of an interface. Junos OS encapsulates the monitored traffic in an IPFIX format and exports the actual packet up to the configured clip length to an collector for further processing. By default, Junos OS supports a maximum clip length of 126 bytes starting from the Ethernet header.

Figure 33 on page 363 illustrates the IPFIX format specification.

Figure 33: Inline Monitoring IPFIX Specification



The IPFIX header and IPFIX payload are encapsulated using IP or UDP transport layer. The exported IPFIX format includes two data records and two data templates that are exported to every collector:

- Data record—Includes incoming and outgoing interface, flow direction, data link frame section, and data link frame size. This information is sent to the collector only when sampled packets are being exported.

  Figure 34 on page 365 is a sample illustration of IPFIX data record packet.

- Option data record—Includes system level information, such as exporting process ID, and sampling interval. This information is sent to the collector periodically, irrespective of whether sampling packets are being exported are not.

  Figure 35 on page 365 is a sample illustration of IPFIX option data record packet.

  Table 50: Information Element fields in IPFIX Option Data Packet

Number	Information Element ID	Information Element Length	Details
1	144	4B	Observation domain ID - An unique identifier of exporting process per IPFIX device. Purpose of this field is to limit the scope of other information element fields.
2	34	4B	Sampling interval at which the packets are sampled. 1000 indicates that one of 1000 packets is sampled.

- Data template—Includes five information elements:
  - Ingress interface
  - Egress interface
  - Flow direction
  - Data link frame size
  - Variable data link frame selection

  Figure 36 on page 366 is a sample illustration of IPFIX data template packet.

- Option data template—Includes flow exporter and sampling interval information.

  Figure 37 on page 366 is a sample illustration of IPFIX option data template packet.

When there is a new or changed inline monitoring services configuration, periodic export of data template and option data template is immediately sent to the respective collectors.

Figure 34: IPFIX Data Record

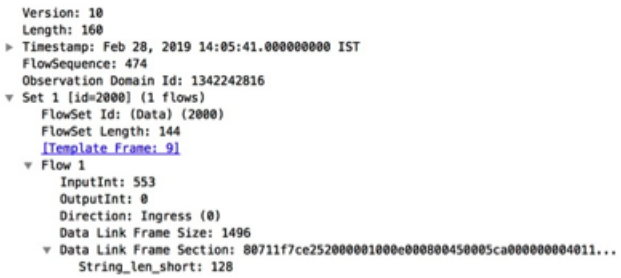


Figure 35: IPFIX Option Data Record

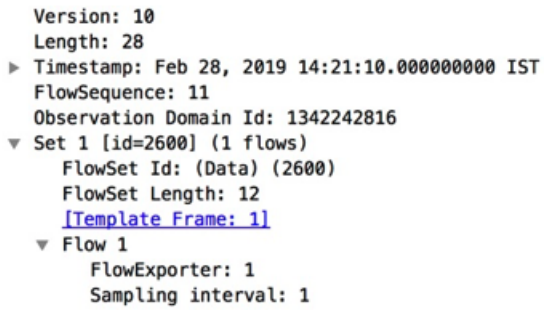


Figure 36: IPFIX Data Template

```
Version: 10
Length: 44
▶ Timestamp: Feb 28, 2019 14:05:42.000000000 IST
FlowSequence: 474
Observation Domain Id: 1342242816
▼ Set 1 [id=2] (Data Template): 2000
   FlowSet Id: Data Template (V10 [IPFIX]) (2)
   FlowSet Length: 28
  ▼ Template (Id = 2000, Count = 5)
      Template Id: 2000
      Field Count: 5
    ▼ Field (1/5): INPUT_SNMP
         0... .... .... .... = Pen provided: No
         .000 0000 0000 1010 = Type: INPUT_SNMP (10)
         Length: 4
    ▼ Field (2/5): OUTPUT_SNMP
         0... .... .... .... = Pen provided: No
         .000 0000 0000 1110 = Type: OUTPUT_SNMP (14)
         Length: 4
    ▼ Field (3/5): DIRECTION
         0... .... .... .... = Pen provided: No
         .000 0000 0011 1101 = Type: DIRECTION (61)
         Length: 1
    ▼ Field (4/5): dataLinkFrameSize
         0... .... .... .... = Pen provided: No
         .000 0001 0011 1000 = Type: dataLinkFrameSize (312)
         Length: 2
    ▼ Field (5/5): dataLinkFrameSection
         0... .... .... .... = Pen provided: No
         .000 0001 0011 1011 = Type: dataLinkFrameSection (315)
         Length: 65535 [i.e.: "Variable Length"]
```

Figure 37: IPFIX Option Data Template

```
Version: 10
Length: 36
▶ Timestamp: Feb 28, 2019 14:21:10.000000000 IST
FlowSequence: 11
Observation Domain Id: 1342242816
▼ Set 1 [id=3] (Options Template): 2600
   FlowSet Id: Options Template (V10 [IPFIX]) (3)
   FlowSet Length: 20
  ▼ Options Template (Id = 2600) (Scope Count = 1; Data Count = 1)
      Template Id: 2600
      Total Field Count: 2
      Scope Field Count: 1
    ▼ Field (1/1) [Scope]: FLOW_EXPORTER
         0... .... .... .... = Pen provided: No
         .000 0000 1001 0000 = Type: FLOW_EXPORTER (144)
         Length: 4
    ▼ Field (1/1): SAMPLING_INTERVAL
         0... .... .... .... = Pen provided: No
         .000 0000 0010 0010 = Type: SAMPLING_INTERVAL (34)
         Length: 4
   Padding: 0000
```

Inline Monitoring Services Configuration Overview

You can configure a maximum of sixteen inline-monitoring instances that support template and collector-specific configuration parameters. Each inline monitoring instance supports up to four collectors (maximum of 64 collectors in total), and you can specify different sampling rates under each collector

configuration. Because of this flexibility, the inline monitoring services overcome the limitations of traditional sampling technologies, such as JFlow, sFlow, and port mirroring.

To configure inline monitoring:

1. You must include the **inline-monitoring** statement at the **[edit services]** hierarchy level. Here you specify the template and inline monitoring instance parameters. You must specify the collector parameters under the inline-monitoring instance.

2. Specify arbitrary match conditions using a firewall filter term and an action to accept the configured inline-monitoring instance. This maps the inline-monitoring instance to the firewall term.

3. Map the firewall filter under the family **inet** or **inet6**. You can also alternatively apply the firewall filter to a forwarding table filter with input or output statement to filter ingress or egress packets, respectively.

Remember:

- The device must support a maximum packet length (clip length) of 126 bytes to enable inline monitoring services.
- You cannot configure more than 16 inline-monitoring instances because of the scarcity of bits available in the packet in the forwarding path.
- Apply inline monitoring services only on a collector interface, that is, the interface on which the collector is reachable. You must not apply inline monitoring on IPFIX traffic as this generates another IPFIX packet for sampling thereby creating a loop. This includes inline monitoring service-generated traffic, such as template and record packet, option template and option record packet.
- When inline monitoring service is enabled on aggregated Ethernet (AE) interfaces, the information element values are as follows:

  Table 51: Information Element Values for Aggregated Ethernet Interfaces

Direction of inline monitoring service on AE interface	Information element-10 (Incoming interface)	Information element-14 (Outgoing interface)
Ingress	SNMP ID of AE	0
Egress	SNMP ID of AE	SNMP ID of member link

- When inline monitoring service is enabled on IRB interfaces, the information element values are as follows:

  Table 52: Information Element Values for IRB Interfaces

Direction of inline monitoring service on IRB interface	Information element-10 (Incoming interface)	Information element-14 (Outgoing interface)

Table 52: Information Element Values for IRB Interfaces *(continued)*

Ingress	SNMP ID of IRB	0
Egress	SNMP ID of IRB	SNMP ID of vlan-bridge encapsulated interface

- For XL-XM based devices (with Lookup chip (XL) and buffering ASIC (XM)), the length of the Data Link Frame Section information element in an exported packet can be shorter than the clip length even if the egress packet length is greater than clip length.

  The length of the Data Link Frame Section information element is reduced by 'N' number of bytes where 'N' = (ingress packet Layer 2 encapsulation length - egress packet Layer 2 encapsulation length).

  For instance, the Layer 2 encapsulation length for the ingress packet is greater than that of the egress packet when the ingress packet has MPLS labels and egress packet is of IPv4 or IPv6 type. When traffic flows from the provider edge (PE) device to the customer edge (CE) device, the ingress packet has VLAN tags and the egress packet is untagged.

  In such cases, the clip length can go past the last address location of the packet head, generating a **PKT_HEAD_SIZE** system log message. This can result in degradation of packet forwarding for the device.

- In case of inline monitoring services in the ingress direction, the **egressInterface** (information element ID 14) does not report SNMP index of the output interface. This information element ID always reports value zero in case of ingress direction. The receiving collector process should identify the validity of this field based on the **flowDirection** (information element ID 61).

Supported and Unsupported Features with Inline Monitoring Services

Inline monitoring services supports:

- Graceful Routing Engine switchover
- In-service software upgrade (ISSU), nonstop software upgrade (NSSU), and nonstop active routing (NSR)
- Ethernet interfaces and integrated routing and bridging (IRB) interfaces
- Junos node slicing

Inline monitoring services currently does not support:

- Ability to configure more than 16 inline-monitoring instances.
- Junos Traffic Vision
- Inline-monitoring-instance action is supported only for **inet** and **inet6** firewall filters. It is not supported for other family filters.
- IPv6 addressable collectors
- Virtual platforms
- Logical systems

Configuring Inline Monitoring Services

SUMMARY

You can configure inline monitoring services to monitor different streams of traffic at different sampling rates on the same logical unit of the interface. You can also export the original packet size to an collector along with information on the interface origin for effective troubleshooting.

The inline monitoring services can monitor both IPv4 and IPv6 traffic on both ingress and egress directions. You can enable inline monitoring on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Before You Configure

When you configure inline monitoring services, you can:

- Configure up to 16 inline-monitoring instances. Under each instance, you can configure specific collector and template parameters.
- Configure up to 4 IPv4-addressable collectors under each inline-monitoring instance. In total, you can configure up to 64 collectors. The collectors can be remote, and at different locations.

  For each collector, you can configure specific parameters, such as source, destination address, sampling rate, forwarding class, and so on. The default routing-instance name at the collector is **default.inet**.
- Configure **inet** or **inet6** family firewall filter with the term action **inline-monitoring-instance** ***inline-monitoring-instance-name***.

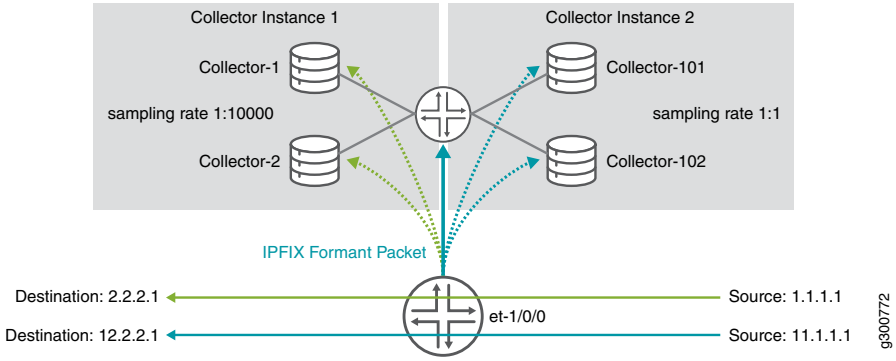
  Each term can support a different inline-monitoring instance.
- Attach the inline monitoring firewall filter under **inet** or **inet6** family of the logical unit of the interface.

After successfully committing the configuration, you can verify the implementation of the inline monitoring services by issuing the **show services inline-monitoring statistics fpc-slot** command from the device CLI.

> **NOTE:** If a packet requires inline monitoring services to be applied along with any of the traditional sampling technologies (such as, JFlow, SFlow, or port mirroring), the Packet Forwarding Engine performs both inline monitoring services and the traditional sampling technology on that packet.

Figure 38 on page 370 is a sample illustration of inline monitoring services, where traffic is monitored at two different sampling rates on the device interface, and exported to four remote collectors in an IPFIX encapsulation format.

Figure 38: Inline Monitoring Services



In this example, the et-1/0/0 interface of the device is configured with inline monitoring services. The details of the configurations are as follows:

- There are two inline-monitoring instances — Instance 1 and Instance 2.
- There are four collectors, two collectors under each inline monitoring instance.
  - Instance 1 has Collector-1 and Collector-2.
  - Instance 2 has Collector-101 and Collector-102.
- The collectors on Instance 1 have a sampling rate of 1:10000.
- The collectors on Instance 2 have a sampling rate of 1:1.
- Instance 1 collectors have a source and destination address of 1.1.1.1 and 2.2.2.1, respectively.
- Instance 2 collectors have a source and destination address of 11.1.1.1 and 12.2.2.1, respectively.
- The packets are exported to the collectors in an IPFIX encapsulated format.

To configure inline monitoring services:

1. Define a firewall filter for each inline-monitoring instance for servicing the inline monitoring services. You can configure **inet** or **inet6** family firewall filter with the term action **inline-monitoring-instance**.

   To define a firewall filter:

   ```
   [edit firewall family family filter filter-name term term]
   user@host# set from source-address source-IPv4-address
   user@host# set from destination-address destination-IPv4-address
   user@host# set then inline-monitoring-instance inline-monitoring-instance-name
   ```

```
user@host# set then action
```

In this example, Terms t1 and t2 are configured for Instance1 and Instance2, respectively.

```
[edit firewall famil inet filter SAMPLE_FOR_1 term t1]
user@host# set from source-address 1.1.1.0/24
user@host# set from destination-address 2.2.2.0/24
user@host# set then inline-monitoring-instance Instance1
user@host# set then accept
user@host# set term t2 from source-address 11.1.1.0/24
user@host# set term t2 from destination-address 12.2.2.0/24
user@host# set term t2 then inline-monitoring-instance Instance2
user@host# set term t2 then accept
```

2. Enable inline monitoring services by configuring the associated template, instance, and collector parameters.

   a. To configure the inline monitoring services template:

   ```
   [edit services inline-monitoring template template-name]
   user@host# set template-refresh-rate template-refresh-rate
   user@host# set option-template-refresh-rate option-template-refresh-rate
   user@host# set observation-domain-id observation-domain-id
   ```

   In this example, templates template-1 and template-2 are configured.

   ```
   [edit services inline-monitoring template template-1]
   user@host# set template-refresh-rate 60
   user@host# set option-template-refresh-rate 100
   user@host# set observation-domain-id 1
   [edit services inline-monitoring template template-2]
   user@host# set template-refresh-rate 60
   user@host# set option-template-refresh-rate 100
   user@host# set observation-domain-id 2
   ```

   b. To configure inline monitoring instance and collector parameters:

   ```
   [edit services inline-monitoring instance inline-monitoring-instance-name]
   user@host# set template-name template-name
   user@host# set maximum-clip-length maximum-clip-length
   user@host# set collector collector-name source-address source-IPv4-address
   user@host# set collector collector-name destination-address destination-IPv4-address
   ```

```
user@host# set collector collector-name destination-port destination-port
user@host# set collector collector-name sampling-rate smapling-rate
```

In this example, Instance1 has two collectors, collector-1 and collector-2, and Instance2 has two collectors, collector-101 and collector-102. Different sampling rates have been configured for both the instances.

```
[edit services inline-monitoring instance Instance1]
user@host# set template-name template-1
user@host# set maximum-clip-length 126
user@host# set collector collector-1 source-address 1.1.1.1
user@host# set collector collector-1 destination-address 2.2.2.1
user@host# set collector collector-1 destination-port 2055
user@host# set collector collector-1 sampling-rate 10000
user@host# set collector collector-2 source-address 1.1.1.1
user@host# set collector collector-2 destination-address 2.2.2.1
user@host# set collector collector-2 destination-port 2055
user@host# set collector collector-2 sampling-rate 10000
```

```
[edit services inline-monitoring instance Instance2]
user@host# set template-name template-2
user@host# set maximum-clip-length 126
user@host# set collector collector-101 source-address 11.1.1.1
user@host# set collector collector-101 destination-address 12.2.2.1
user@host# set collector collector-101 destination-port 2055
user@host# set collector collector-101 sampling-rate 1
user@host# set collector collector-102 source-address 11.1.1.1
user@host# set collector collector-102 destination-address 2.2.2.1
user@host# set collector collector-102 destination-port 2055
user@host# set collector collector-102 sampling-rate 1
```

3. Map the firewall filter under the family **inet** or **inet6** of the logical unit of the interface to apply inline monitoring in the ingress or egress direction.

   Alternatively, you can apply inline monitoring by mapping the firewall filter to a forwarding table filter with input or output statement to filter ingress or egress packets, respectively.

   To attach the firewall filter:

   ```
   [edit interfaces interface-name]
   user@host# set unit 0 family family filter input filter
   user@host# set unit 0 family family address ip-address
   ```

In this example, the inline monitoring filter is attached to family inet of unit 0 of et-1/0/0.

```
[edit interfaces et-1/0/0]
user@host# set unit 0 family inet filter input SAMPLE_FOR_1
user@host# set unit 0 family inet address 10.100.0.1/30
```

4

PART

Sampling, Discard Accounting, and Port Mirroring Services

CHAPTER 10

Sampling Data Using Traffic Sampling and Discard Accounting

IN THIS CHAPTER

Configuring Traffic Sampling on MX, M and T Series Routers

IN THIS SECTION

Traffic sampling enables you to copy traffic to a Physical Interface Card (PIC) that performs flow accounting while the router forwards the packet to its original destination. You can configure the router to perform sampling in one of the following three locations:

- On the Routing Engine, using the sampled process. To select this method, use a filter (input or output) with a matching term that contains the **then sample** statement.
- On the Monitoring Services, Adaptive Services, or Multiservices PIC.

- On an inline data path without the need for a services Dense Port Concentrator (DPC). To do this inline active sampling, you define a sampling instance with specific properties. One Flexible PIC Concentrator (FPC) can support only one instance; for each instance, either services PIC-based sampling or inline sampling is supported per family. Inline sampling supports version 9 and IPFIX flow collection templates.

> **NOTE:** Routing Engine based sampling is not supported on VPN routing and forwarding (VRF) instances.

The following sections provide configuration instructions for traffic sampling:

Configuring Firewall Filter for Traffic Sampling

To configure firewall filter for traffic sampling, you must perform the following tasks:

- Create a firewall filter to apply to the logical interfaces being sampled by including the **filter** statement at the **[edit firewall family *family-name*]** hierarchy level. In the filter **then** statement, you must specify the action modifier **sample** and the action **accept**.

```
filter filter-name {
    term term-name {
        then {
            sample;
            accept;
        }
    }
}
```

For more information about firewall filter actions and action modifiers, see the *Routing Policies, Firewall Filters, and Traffic Policers User Guide*.

- Apply the filter to the interfaces on which you want to sample traffic by including the **address** and **filter** statements at the **[edit interfaces *interface-name* unit *logical-unit-number* family *family-name*]** hierarchy level:

```
address address {
}
filter {
    input filter-name;
}
```

The following prerequisites apply to M, MX, and T Series routers when you configure traffic sampling on interfaces and in firewall filters:

- If you configure a sample action in a firewall filter for an inet or inet6 family on an interface without configuring the forwarding-options settings, operational problems might occur if you also configure port mirroring or flow-tap functionalities. In such a scenario, all the packets that match the firewall filter are incorrectly sent to the service PIC.
- If you include the **then sample** statement at the **[edit firewall family inet filter *filter-name* term *term-name*]** hierarchy level to specify a sample action in a firewall filter for IPv4 packets, you must also include the **family inet** statement at the **[edit forwarding-options sampling]** hierarchy level or the **instance *instance-name* family inet** statement at the **[edit forwarding-options sampling]** hierarchy level. Similarly, if you include the then **sample** statement at the **[edit firewall family inet6 filter *filter-name* term *term-name*]** hierarchy level to specify a sample action in a firewall filter for IPv6 packets, you must also include **family inet6** statement at the **[edit forwarding-options sampling]** hierarchy level or the **instance *instance-name* family inet6** statement at the **[edit forwarding-options sampling]** hierarchy level. Otherwise, a commit error occurs when you attempt to commit the configuration.
- Also, if you configure traffic sampling on a logical interface by including the sampling input or sampling output statements at the **[edit interface *interface-name* unit *logical-unit-number*]** hierarchy level, you must also include the **family inet | inet6** statement at the **[edit forwarding-options sampling]** hierarchy level, or the **instance *instance-name* family inet | inet6** statement at the **[edit forwarding-options sampling]** hierarchy level.

Configuring Traffic Sampling on a Logical Interface

To configure traffic sampling on any logical interface, enable sampling and specify a non zero sampling rate by including the sampling statement at the **[edit forwarding-options]** hierarchy level:

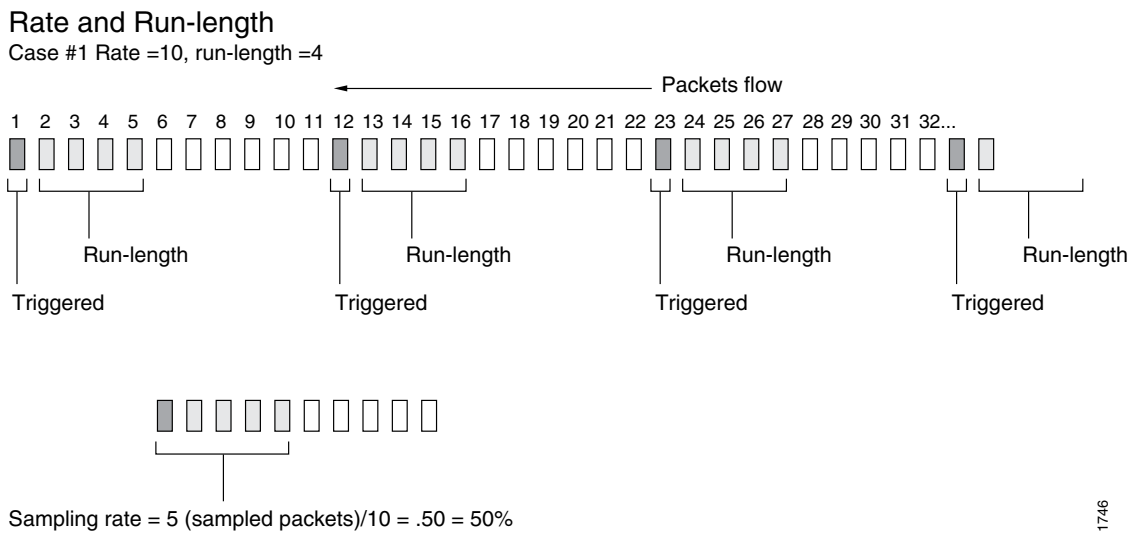
```
sampling {
    input {
        rate number;
        run-length number;
        max-packets-per-second number;
        maximum-packet-length bytes;
    }
```

When you use Routing Engine-based sampling, specify the threshold traffic value by including the **max-packets-per-second** statement. The value is the maximum number of packets to be sampled, beyond which the sampling mechanism begins dropping packets. The range is from 0 through 65,535. A value of 0 instructs the Packet Forwarding Engine not to sample any packets. The default value is 1000.

> **NOTE:** When you configure active monitoring and specify a Monitoring Services, Adaptive Services, or Multiservices PIC in the **output** statement, or when you configure inline sampling, the **max-packets-per-second** value is ignored.

Specify the sampling rate by setting the values for **rate** and **run-length** (see Figure 39 on page 378).

Figure 39: Configuring Sampling Rate



> **NOTE:** Do not configure ingress sampling on **ms-** logical interfaces on which PIC-based flow monitoring is enabled, which causes undesired flow monitoring behavior and might result in repeated sampling of a single packet. Starting in Junos OS Release 15.1, a commit error occurs when you try to configure ingress traffic sampling on that interface. In Junos OS Release 14.2 and earlier, the commit error does not occur, but you should not configure ingress traffic sampling on that interface.
>
> If PIC-based flow monitoring is enabled on an ms-*fpc/pic/port.logical-unit* interface, a commit check error occurs when you attempt to configure ingress traffic sampling on that interface. This error occurs because a combination of ingress sampling and PIC-based flow monitoring operations on an **ms-** logical interface causes undesired flow monitoring behavior and might result in repeated sampling of a single packet. You must not configure ingress sampling on **ms-** logical interfaces on which PIC-based flow monitoring is enabled.

The **rate** statement specifies the ratio of packets to be sampled. For example, if you configure a rate of 10, *x* number of packets out of every 10 is sampled, where *x*=run length + 1. By default, the rate is 0, which means that no traffic is sampled.

The **run-length** statement specifies the number of matching packets to sample following the initial one-packet trigger event. By default, the run length is 0, which means that no more traffic is sampled after the trigger event. The range is from 0 through 20. Configuring a run length greater than 0 allows you to sample packets following those already being sampled.

> **NOTE:** The **run-length** and **maximum-packet-length** configuration statements are not supported on MX80 routers.

If you do not include the **input** statement, sampling is disabled.

To collect the sampled packets in a file, include the **file** statement at the **[edit forwarding-options sampling output]** hierarchy level. Output file formats are discussed later in the chapter.

Disabling Traffic Sampling

To explicitly disable traffic sampling on the router, include the **disable** statement at the **[edit forwarding-options sampling]** hierarchy level:

```
disable;
```

Sampling Once

To explicitly sample a packet for active monitoring only once, include the **sample-once** statement at the **[edit forwarding-options sampling]** hierarchy level:

```
sample-once;
```

Setting this option avoids duplication of packets in cases where sampling is enabled at both the ingress and egress interfaces and simplifies analysis of the sampled traffic.

Preserving Prerewrite ToS Value for Egress Sampled or Mirrored Packets

Starting in Junos OS Release 14.1, you can preserve the prenormalized type-of-service (ToS) value in egress sampled or mirrored packets. Include the **pre-rewrite-tos** statement at the **[edit forwarding-options sampling]** hierarchy level.

On MPC-based interfaces, you can configure ToS rewrite either using class-of-service (CoS) configuration by including the **rewrite-rules dscp *rule_name*** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level or using firewall filter configuration by including the **dscp** statement at the **[edit firewall family *family-name* filter *filter-name* term *term-name* then]** hierarchy level. If ToS rewrite is configured, the egress mirrored or sampled copies contain the post-rewrite ToS values by default. With the **pre-rewrite-tos** configuration, you can retain the prerewrite ToS value in the sampled or mirrored packets.

NOTE:

- If ToS rewrite is configured on the egress interface by using both CoS and firewall filter configuration, and if the **pre-rewrite-tos** statement is also configured, then the egress sampled packets contain the DSCP value set using the firewall filter configuration. However, if the **pre-rewrite-tos** statement is not configured, the egress sampled packets contain the DSCP value set by the CoS configuration.
- With the **pre-rewrite-tos** statement, you can configure retaining prenormalization ToS values only for sampling done under **family inet** and **family inet6**.
- This feature cannot be configured at the **[edit logical-systems]** hierarchy level. It can be configured only at the global level under the **forwarding-option** configuration.
- When ToS rewrite is configured by using a firewall filter on both ingress and egress interfaces, the egress sampled packets contain the DSCP value set by the ingress ToS rewrite configuration if the **pre-rewrite-tos** statement is configured. However, if the **pre-rewrite-tos** statement is not configured, the egress sampled packets contain the DSCP value set by the ToS rewrite configuration for the egress firewall filter.
- If the **pre-rewrite-tos** statement is configured, and a deactivate or delete operation is performed at the **[edit forwarding-options]** hierarchy level, **pre-rewrite-tos** configuration still remains active. To disable the **pre-rewrite-tos** configuration for such a case, you must explicitly deactivate or delete the **pre-rewrite-tos** statement at the **[edit forwarding-options sampling]** hierarchy level before performing a deactivate or delete operation at the **[edit forwarding-options]** hierarchy level.

Configuring Traffic Sampling Output

To configure traffic sampling output, include the following statements at the **[edit forwarding-options sampling family (inet | inet6 | mpls) output]** hierarchy level:

```
aggregate-export-interval seconds;
flow-active-timeout seconds;
flow-inactive-timeout seconds;
extension-service service-name;
  flow-server hostname {
    aggregation {
      autonomous-system;
      destination-prefix;
      protocol-port;
      source-destination-prefix {
        caida-compliant;
```

```
        }
        source-prefix;
    }
    autonomous-system-type (origin | peer);
    (local-dump | no-local-dump);
    port port-number;
    source-address address;
    version format;
    version9 {
        template template-name;
    }
}
interface interface-name {
    engine-id number;
    engine-type number;
    source-address address;
}
file {
    disable;
    filename filename;
    files number;
    size bytes;
    (stamp | no-stamp);
    (world-readable | no-world-readable);
}
```

To configure inline flow monitoring on MX Series routers, include the **inline-jflow** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls) output]** hierarchy level. Inline sampling exclusively supports a new format called IP_FIX that uses UDP as the transport protocol. When you configure inline sampling, you must include the **version-ipfix** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls) output flow-server *address*]** hierarchy level and also at the **[edit services flow-monitoring]** hierarchy level. For more information about configuring inline flow monitoring, see "Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250" on page 63.

To direct sampled traffic to a flow-monitoring interface, include the **interface** statement. The **engine-id** and **engine-type** statements specify the identity and type numbers of the interface; they are dynamically generated based on the Flexible PIC Concentrator (FPC), PIC, and slot numbers and the chassis type. The **source-address** statement specifies the traffic source.

Starting in Junos OS Release 19.3R1, to configure inline flow monitoring on Juniper Sky Advanced Threat Prevention (ATP), include the **flow-server** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls) output]** hierarchy level. Inline sampling exclusively supports a new format called IP_FIX that uses UDP as the transport protocol. When you configure inline sampling,

you must include the **version-ipfix** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls) output flow-server *address*]** hierarchy level and also at the **[edit services flow-monitoring]** hierarchy level.

To configure flow sampling version 9 output, you need to include the **template** statement at the **[edit forwarding-options sampling output version9]** hierarchy level. For information on cflowd, see "Enabling Flow Aggregation" on page 489.

The **aggregate-export-interval** statement is described in "Configuring Discard Accounting" on page 390, and the **flow-active-timeout** and **flow-inactive-timeout** statements are described in "Configuring Flow Monitoring" on page 3.

Traffic sampling results are automatically saved to a file in the **/var/tmp** directory. To collect the sampled packets in a file, include the **file** statement at the **[edit forwarding-options sampling family inet output]** hierarchy level:

```
file {
    disable;
    filename filename;
    files number;
    size bytes;
    (stamp | no-stamp);
    (world-readable | no-world-readable);
}
```

Traffic Sampling Output Format

Traffic sampling output is saved to an ASCII text file. The following is an example of the traffic sampling output that is saved to a file in the **/var/tmp** directory. Each line in the output file contains information for one sampled packet. You can optionally display a timestamp for each line.

The column headers are repeated after each group of 1000 packets.

```
# Apr  7 15:48:50
Time                    Dest           Src Dest Src Proto TOS Pkt Intf  IP   TCP
                        addr          addr port port          len num frag flags
Apr 7 15:48:54 192.168.9.194 192.168.9.195   0    0   1   0x0  84  8   0x0   0x0
Apr 7 15:48:55 192.168.9.194 192.168.9.195   0    0   1   0x0  84  8   0x0   0x0
Apr 7 15:48:56 192.168.9.194 192.168.9.195   0    0   1   0x0  84  8   0x0   0x0
Apr 7 15:48:57 192.168.9.194 192.168.9.195   0    0   1   0x0  84  8   0x0   0x0
Apr 7 15:48:58 192.168.9.194 192.168.9.195   0    0   1   0x0  84  8   0x0   0x0
```

To set the timestamp option for the file **my-sample**, enter the following:

```
[edit forwarding-options sampling output file]
user@host# set filename my-sample files 5 size 2m world-readable stamp;
```

Whenever you toggle the timestamp option, a new header is included in the file. If you set the **stamp** option, the **Time** field is displayed.

```
# Apr  7 15:48:50
# Time            Dest        Src  Dest   Src Proto  TOS   Pkt  Intf    IP   TCP
#                 addr       addr  port  port              len   num  frag flags
# Feb  1 20:31:21
#                 Dest        Src  Dest   Src Proto  TOS   Pkt  Intf    IP   TCP
#                 addr       addr  port  port              len   num  frag flags
```

Tracing Traffic Sampling Operations

Tracing operations track all traffic sampling operations and record them in a log file in the **/var/log** directory. By default, this file is named **/var/log/sampled**. The default file size is 128K, and 10 files are created before the first one gets overwritten.

To trace traffic sampling operations, include the **traceoptions** statement at the **[edit forwarding-options sampling]** hierarchy level:

```
traceoptions {
  no-remote-trace;
  file filename <files number> <size bytes> <match expression> <world-readable | no-world-readable>;
}
```

Traffic Sampling Examples

IN THIS SECTION

The following sections provide examples of configuring traffic sampling:

Example: Sampling a Single SONET/SDH Interface

The following configuration gathers statistical sampling information from a small percentage of all traffic on a single SONET/SDH interface and collects it in a file named **sonet-samples.txt**.

Create the filter:

```
[edit firewall family inet]
filter {
    input sample-sonet {
        then {
            sample;
            accept;
        }
    }
}
```

Apply the filter to the SONET/SDH interface:

```
[edit interfaces]
so-0/0/1 {
    unit 0 {
        family inet {
            filter {
                input sample-sonet;
            }
            address 10.127.68.254/32 {
                destination 172.16.74.7;
            }
        }
    }
}
```

Finally, configure traffic sampling:

```
[edit forwarding-options]
sampling {
    input {
        family inet {
            rate 100;
            run-length 2;
        }
    }
    family inet {
        output {
```

```
        file {
            filename sonet-samples.txt;
            files 40;
            size 5m;
        }
    }
  }
}
```

Example: Sampling All Traffic from a Single IP Address

The following configuration gathers statistical information about every packet entering the router on a specific Gigabit Ethernet port originating from a single source IP address of **172.16.92.31**, and collects it in a file named **samples-172-16-92-31.txt**.

Create the filter:

```
[edit firewall family inet]
filter one-ip {
  term get-ip {
    from {
      source-address 172.16.92.31;
    }
    then {
      sample;
      accept;
    }
  }
}
```

Apply the filter to the Gigabit Ethernet interface:

```
[edit interfaces]
ge-4/1/1 {
  unit 0 {
    family inet {
      filter {
        input one-ip;
      }
      address 10.45.92.254;
    }
  }
}
```

Finally, gather statistics on all the candidate samples; in this case, gather all statistics:

```
[edit forwarding-options]
sampling {
    input {
        family inet {
            rate 1;
        }
    }
    family inet {
        output {
            file {
                filename samples-172-16-92-31.txt;
                files 100;
                size 100k;
            }
        }
    }
}
```

Example: Sampling All FTP Traffic

The following configuration gathers statistical information about a moderate percentage of packets using the FTP data transfer protocol in the output path of a specific T3 interface, and collects the information in a file named **t3-ftp-traffic.txt**.

Create a filter:

```
[edit firewall family inet]
filter ftp-stats {
    term ftp-usage {
        from {
            destination-port [ftp ftp-data];
        }
        then {
            sample;
            accept;
        }
    }
}
```

Apply the filter to the T3 interface:

```
[edit interfaces]
t3-7/0/2 {
    unit 0 {
```

```
        family inet {
            filter {
                input ftp-stats;
            }
            address 10.35.78.254/32 {
                destination 10.35.78.4;
            }
        }
    }
}
```

Finally, gather statistics on 10 percent of the candidate samples:

```
[edit forwarding-options]
sampling {
    input {
        family inet {
            rate 10;
        }
    }
    family inet {
        output {
            file {
                filename t3-ftp-traffic.txt;
                files 50;
                size 1m;
            }
        }
    }
}
```

Release History Table

Release	Description
14.1	Starting in Junos OS Release 14.1, you can preserve the prenormalized type-of-service (ToS) value in egress sampled or mirrored packets. Include the **pre-rewrite-tos** statement at the **[edit forwarding-options sampling]** hierarchy level.

RELATED DOCUMENTATION

Traffic Sampling, Forwarding, and Monitoring Overview

Configuring Sampling Instance on MX, M and T Series Routers or QFX Series Switches

You can configure active sampling by defining a sampling instance that specifies a name for the sampling parameters and bind the instance name to an FPC, MPC, or DPC. This configuration enables you to define multiple named sampling parameter sets associated with multiple destinations and protocol families per sampling destination. With the cflowd version 5 and version 8 and flow aggregation version 9, you can use templates to organize the data gathered from sampling.

To implement this feature, you include the **instance** statement at the **[edit forwarding-options sampling]** hierarchy level.

The following considerations apply to the sampling instance configuration:

- This configuration is supported on the IP version 4 (**inet**), IP version 6 (**ipv6**), and MPLS protocol families.
- You can configure the router to perform sampling in either of two locations:
  - On the Routing Engine, using the sampled process. To select this method, use a filter (input or output) with a matching term that contains the **then** sample statement.
  - On the Monitoring Services, Adaptive Services, or Multiservices PIC. Specify the interface name at the **[forwarding-options sampling instance *instance-name* family inet output interface]** hierarchy level. You can configure the same or different services PICs in a set of sampling instances.
- You can configure the **rate** and **run-length** options at the **[edit forwarding-options sampling input]** hierarchy level to apply common values for all families on a global basis. Alternatively, you can configure these options at the **[edit forwarding-options sampling instance *instance-name* input]** hierarchy level to apply specific values for each instance or at the **[edit forwarding-options sampling instance *instance-name* family *family* input]** hierarchy level to apply specific values for each protocol family you configure.
- Starting in Junos OS Release 16.1, for inline active flow monitoring, you can configure a Differentiated Services Code Point (DSCP) mapping and a forwarding class to apply to exported packets. Use the **dscp** and **forwarding-class** options at the **[edit forwarding-options sampling *instance-name* family (inet | inet6) output flow-server *hostname*]** hierarchy level.
- For MX Series devices with Modular Port Concentrators (MPCs), port-mirrored or sampled packets can be truncated (or clipped) to any length in the range of 1 through 255 bytes. Only the values 1 to 255 are valid for packet truncation on these devices. For other devices, the range is from 0 through 9216. A maximum-packet-length value of zero (0) represents that truncation is disabled, and the entire packet is mirrored or sampled.

> **NOTE:** The **run-length** and **maximum-packet-length** configuration statements are not supported on MX80 routers.

To associate the defined instance with a particular FPC, MPC, or DPC, you include the **sampling-instance** statement at the **[edit chassis fpc *number*]** hierarchy level, as in the following example:

```
chassis {
    fpc 2 {
        sampling-instance samp1;
    }
}
```

Starting in Junos OS Release 14.1, you can associate a sampling instance with an FPC in the MX Series Virtual Chassis master or backup router. Use the **sampling-instance *instance-name*** statement at the **[edit chassis member *member-number* fpc slot *slot-number*]** hierarchy level, where *member-number* is 0 (for the master router) or 1 (for the backup router), and *slot-number* is a number in the range 0 through 11.

Release History Table

Release	Description
16.1	Starting in Junos OS Release 16.1, for inline active flow monitoring, you can configure a Differentiated Services Code Point (DSCP) mapping and a forwarding class to apply to exported packets.
14.1	Starting in Junos OS Release 14.1, you can associate a sampling instance with an FPC in the MX Series Virtual Chassis master or backup router.

RELATED DOCUMENTATION

- *Traffic Sampling, Forwarding, and Monitoring Overview*
- *Monitoring, Sampling, and Collection Services Interfaces User Guide*
- Configuring Active Flow Monitoring | **34**
- *Directing Traffic Sampling Output to a Server Running the cflowd Application*
- Configuring Traffic Sampling on MX, M and T Series Routers | **375**
- Example: Sampling Instance Configuration | **118**
- sampling (Forwarding Options) | **1124**
- *Inline Flow Monitoring for Virtual Chassis Overview*

Configuring Discard Accounting

Discard accounting is similar to traffic sampling, but varies from it in two ways:

- In discard accounting, the packet is intercepted by the monitoring PIC and is not forwarded to its destination.
- Traffic sampling allows you to limit the number of packets sampled by configuring the **max-packets-per-second**, **rate**, and **run-length** statements. Discard accounting does not provide these options, and a high packet count can potentially overwhelm the monitoring PIC.

A discard instance is a named entity that specifies collector information under the **accounting *name*** statement. Discard instances are referenced in firewall filter **term** statements by including the **then discard accounting *name*** statement.

Most of the other statements are also found at the **[edit forwarding-options sampling]** hierarchy level. For information on cflowd, see "Enabling Flow Aggregation" on page 489. The **flow-active-timeout** and **flow-inactive-timeout** statements are described in "Configuring Flow Monitoring" on page 3.

To direct sampled traffic to a flow-monitoring interface, include the **interface** statement. The **engine-id** and **engine-type** statements specify the accounting interface used on the traffic, and the **source-address** statement specifies the traffic source.

You cannot use rate-limiting with discard accounting; however, you can specify the duration of the interval for exporting aggregated accounting information by including the **aggregate-export-interval** statement in the configuration. This enables you to put a boundary on the amount of traffic exported to a flow-monitoring interface.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

Configuring Flow Monitoring | 3

CHAPTER 11

Sampling Data Using Inline Sampling

IN THIS CHAPTER

Understanding Inline Active Flow Monitoring

Inline active flow monitoring is implemented on the Packet Forwarding Engine. The Packet Forwarding Engine performs functions such as creating flows, updating flows, and exporting flow records to a flow collector. The flow records are sent out in industry-standard IPFIX or version 9 format. IPFIX and version 9 templates use UDP as the transport protocol.

You can configure inline active flow monitoring for IPv4, IPv6, MPLS, MPLS-IPv4, VPLS, and bridge traffic. Starting in Junos OS Release 18.1R1, you can configure inline active flow monitoring for MPLS-over-UDP traffic for PTX Series routers, except for the PTX10002-60C. Starting in Junos OS Release 18.2R1, you can configure inline active flow monitoring for MPLS, MPLS-IPv4, and MPLS-IPv6 traffic for PTX Series routers. Starting in Junos OS Release 18.2R1, you can configure inline active flow monitoring for bridge traffic for MX Series routers.

Starting in Junos OS Release 18.4R1, you can configure inline active flow monitoring for MPLS-IPv6 traffic for MX Series routers.

Starting with Junos OS Release 19.4R1 on the PTX10002-60C router, you can perform flow monitoring for MPLS-over-UDP flows to look past the tunnel header to sample and report on the inner payload at both the transit and egress nodes of the tunnel. MPLS IPv4 and IPv6 payloads and IPFIX and version 9 templates are supported. Only ingress sampling is supported.

Benefits of Inline Active Flow Monitoring

Inline active flow monitoring is implemented on the Packet Forwarding Engine rather than on a services card. This enables:

- Lower cost—You do not need to invest in additional hardware.
- Higher scalabilty—You do not need to dedicate a PIC slot for a services PIC, so you can make full use of the available slots for handling traffic on the device.
- Better performance—Inline flow monitoring performance is not dependent on the capacity of a services card.

Inline Active Flow Monitoring Configuration Overview

The inline active flow monitoring configuration can be broadly classified into four categories:

1. Configurations at the **[edit services flow-monitoring]** hierarchy level—At this level, you configure the template properties for inline flow monitoring.
2. Configurations at the **[edit forwarding-options]** hierarchy level—At this level, you configure a sampling instance and associate the template (configured at the **[edit services flow-monitoring]** hierarchy level) with the sampling instance. At this level, you also configure the flow-server IP address and port number as well as the flow export rate, and specify the collectors.

   You cannot change the source IP address for collectors under the same family. Also, the template mapped across collectors under a family should be the same.
3. Configurations at the **[edit chassis]** hierarchy level—At this level, you associate the sampling instance with the FPC on which the media interface is present. If you are configuring sampling of IPv4 flows, IPv6 flows, or VPLS flows, you can configure the flow hash table size for each family.
4. Configurations at the **[edit firewall]** hierarchy level—At this level you configure a firewall filter for the family of traffic to be sampled. You must attach this filter to the interface on which you want to sample the traffic.

Before you configure inline active flow monitoring, ensure that you have adequately-sized hash tables for IPv4, IPv6, MPLS, and VPLS flow sampling. These tables can use from one up to fifteen 256K areas. Starting with Junos OS Release 16.1R1 and 15.1F2, the IPv4 table is assigned a default value of 1024. Prior to Junos OS Release 16.1 and 15.1F2, the IPv4 table is assigned a default value of fifteen 256K areas. The IPv6 table is assigned a default value of 1024, and the VPLS table is assigned a default value of 1024. Allocate larger tables when anticipated traffic volume makes it necessary.

You can configure flow collectors to be reachable through non-default VPN routing and forwarding (VRF) instances by including the **routing-instance instance-name** statement at the **[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls) output flow-server hostname]** hierarchy level for inline flow monitoring. You cannot configure a flow collector to be reachable through non-default VRF

instances for version 5 and version 8 flows. You must configure the routing instance to be a VRF instance by including the **instance-type vrf** statement at the **[edit routing-instances instance-name]** hierarchy level.

Inline Active Flow Monitoring Limitations and Restrictions

The following limitations and restrictions apply to the inline active flow monitoring feature in Junos OS:

- Inline active flow monitoring is not supported for input or output traffic on MS-MPC or MS-MIC-16G interfaces.
- In Junos OS release 15.1 and earlier, you can apply version 9 flow templates to IPv4 traffic. Starting in Junos OS Release 16.1, you can also apply version 9 flow templates to MPLS and MPLS-IPv4 traffic. Starting in Junos OS Release 18.1R1, you can also apply version 9 flow templates to IPv6 traffic.
- In Junos OS Release 15.1 and earlier, you can apply IPFIX flow templates to IPv4, IPv6, and VPLS traffic. Starting in Junos OS release 16.1, you can also apply IPFIX flow templates to MPLS and MPLS-IPv4 traffic.
- Starting with Junos OS Release 17.2R1, you can apply IPFIX flow templates to unicast IPv4 and IPv6 traffic on QFX10002 switches. Starting with Junos OS Release 17.4R1, you can apply IPFIX flow templates to unicast IPv4 and IPv6 traffic on QFX10008 and QFX10016 switches.
- Inline active flow monitoring is not supported when you enable Next Gen Services on an MX Series router.
- You can configure only one sampling instance on a Flexible PIC Concentrator (FPC).
- You can configure only one type of sampling—either services-card-based sampling or inline sampling—per family in a sampling instance. However, you can configure services-card-based and inline sampling for different families in a sampling instance.
- The following considerations apply to the inline sampling instance configuration:
  - Sampling run-length and clip-size are not supported.
  - In Junos OS Release 16.2 and in Junos OS Release 16.1R3 and earlier, you can configure only one collector under a family for inline active flow monitoring. Starting with Junos OS Release 16.1R4 and 17.2R1, you can configure up to four collectors under a family for inline active flow monitoring.
  - The user-defined sampling instance gets precedence over the global instance. When a user-defined sampling instance is attached to the FPC, the global instance is removed from the FPC and the user-defined sampling instance is applied to the FPC.
- Flow records and templates cannot be exported if the flow collector is reachable through any management interface.
- If the destination of the sampled flow is reachable through multiple paths, the IP_NEXT_HOP (Element ID 15) and OUTPUT_SNMP (Element ID 14) in the IPv4 and IPv6 flow records are not reported correctly unless you enable learning of next hop addresses by using the **nexthop-learning enable** statement. If you do *not* use **nexthop-learning enable**:

  - For IPv4 flow records, the IP_NEXT_HOP and OUTPUT_SNMP are set to the Gateway Address and SNMP Index of the first path seen in the forwarding table.
  - For IPv6 flow records, the IP_NEXT_HOP and OUTPUT_SNMP are set to 0.
- The Incoming Interface (IIF) and Outgoing Interface (OIF) should be part of the same VRF. If OIF is in a different VRF, DST_MASK (Element ID 13), DST_AS (Element ID 17), IP_NEXT_HOP (Element ID 15), and OUTPUT_SNMP (Element ID 14) are set to 0 in the flow records.
- Each lookup chip maintains and exports flows independent of other lookup chips. Traffic received on a media interface is distributed across all lookup chips in a multilookup chip platform. It is likely that a single flow is processed by multiple lookup chips. Therefore, each lookup chip creates a unique flow and exports it to the flow collector. This can cause duplicate flow records to go to the flow collector. The flow collector should aggregate PKTS_COUNT and BYTES_COUNT for duplicate flow records to derive a single flow record.

IPFIX and Version 9 Templates

IN THIS SECTION

The following sections list the fields included in IPFIX and version 9 templates.

Fields Included in the IPFIX Bridge Template for MX Series

Table 53 on page 396 shows the fields that are included in the IPFIX Bridge template. The fields are shown in the order in which they appear in the template.

Table 53: IPFIX Bridge Template Fields for MX, M, and T Series

Field	Element ID
Destination MAC	80
Source MAC	56
Ethernet Type	256
Input SNMP	10
Output SNMP	14
Flow End Reason	136
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX IPv4 Template for MX, M, and T Series

Table 54 on page 396 shows the fields that are included in the IPFIX IPv4 template. The fields are shown in the order in which they appear in the template.

Table 54: IPFIX IPv4 Template Fields for MX, M, and T Series

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 ToS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32

Table 54: IPFIX IPv4 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Input Interface	10
VLAN ID	58
IPv4 Source Mask	9
IPv4 Destination Mask	13
Source AS	16
Destination AS	17
IPv4 Next Hop Address	15
TCP Flags	6
Output Interface	14
Minimum TTL	52
Maximum TTL	53
Flow End Reason	136
IP Protocol Version	60
BGP IPv4 Next Hop Address	18
Flow Direction (Starting in Junos OS Release 16.1)	61
802.1Q VLAN identifier (dot1qVlanId)	243
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152

Table 54: IPFIX IPv4 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX IPv4 Template for PTX Series

Table 55 on page 398 shows the fields that are available in the IPFIX IPv4 template.

Table 55: IPFIX IPv4 Template Fields for PTX Series

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 TOS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
Source AS	16
Destination AS	17
BGP IPv4 Next Hop Address	18
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Table 55: IPFIX IPv4 Template Fields for PTX Series *(continued)*

Field	Element ID
IPv4 Next Hop Address	15
IPv4 Source Mask	9
IPv4 Destination Mask	13
TCP Flags	6
IP Protocol Version	60
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153
The type of interface where packets are being received. This field can have the following values: • 1—Other (default value) • 131—De-encapsulated GRE traffic is reported as *tunnel*	368

Fields Included in the IPFIX IPv4 Template for PTX10003-160C, PTX10003-80C, and PTX10008 with JNP10K-LC1201 line card) routers

Table 56 on page 399 shows the fields that are available in the IPFIX IPv4 template.

Table 56: IPFIX IPv4 Template Fields for PTX10003-160C, PTX10003-80C, and PTX10008 routers

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 TOS	5
IPv4 Protocol	4
Source Port	7
Destination Port	11
ICMP Type and Code	32

Table 56: IPFIX IPv4 Template Fields for PTX10003-160C, PTX10003-80C, and PTX10008 routers *(continued)*

Field	Element ID
Output SNMP Index	14
Input SNMP Index	10
IPv4 Source Mask	9
IPv4 Destination Mask	13
Source AS	16
Destination AS	17
IPv4 Next Hop	15
BGP Next Hop Address	18
Time the flow started with respect to Epoch time	152
TCP Flags	6
Number of Bytes	1
Number of Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IP Protocol Version	60
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX IPv6 Template for MX, M, and T Series

Table 57 on page 400 shows the fields that are included in the IPFIX IPv6 template. The fields are shown in the order in which they appear in the template.

Table 57: IPFIX IPv6 Template Fields for MX, M, and T Series

Field	Element ID
IPv6 Source Address	27

Table 57: IPFIX IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
IPv6 Destination Address	28
IPv6 ToS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	139
Input Interface	10
VLAN ID	58
IPv6 Source Mask	29
IPv6 Destination Mask	30
Source AS	16
Destination AS	17
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
TCP Flags	6
Output Interface	14
Minimum Hop Limits	52
Maximum Hop Limits	53
Flow End Reason	136
Flow Direction (Starting in Junos OS Release 16.1)	61
802.1Q VLAN identifier (dot1qVlanId)	243

Table 57: IPFIX IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
IPv6 Option Headers	64
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX IPv6 Template for PTX Series

Table 63 on page 410 shows the fields that are available in the IPv6 templates.

Table 58: IPFIX IPv6 Template Fields for PTX Series

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 TOS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	139
Input Interface	10
IPv6 Source Mask	29
IPv6 Destination Mask	30

Table 58: IPFIX IPv6 Template Fields for PTX Series *(continued)*

Field	Element ID
TCP Flags	6
Source AS	16
Destination AS	17
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
IP Protocol Version	60
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153
The type of interface where packets are being received. This field can have the following values: • 1—Other (default value) • 131—De-encapsulated GRE traffic is reported as *tunnel*	368

Fields Included in the IPFIX IPv6 Template for PTX10003-160C, PTX10003-80C, and PTX10008 (with JNP10K-LC1201 line card) routers

Table 59 on page 403 shows the fields that are available in the IPv6 templates.

Table 59: IPFIX IPv6 Template Fields for PTX10003-160C, PTX10003-80C, and 10008 routers

Field	Element ID
IPv6 Source Address	27

Table 59: IPFIX IPv6 Template Fields for PTX10003-160C, PTX10003-80C, and 10008 routers *(continued)*

Field	Element ID
IPv6 Destination Address	28
IPv6 TOS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	139
Time the flow ended with respect to Epoch time	153
Input SNMP Index	10
Output SNMP Index	14
IPv6 Source Mask	29
IPv6 Destination Mask	30
Source AS	16
Destination AS	17
IPv6 Next Hop Address	62
TCP Flags	6
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IP Protocol Version	60
Time the flow started with respect to Epoch time	152

Fields Included in the IPFIX MPLS-IPv4 Template for MX, M, and T Series

Starting in Junos OS Release 16.1, the IPFIX MPLS-IPv4 template is supported. Table 60 on page 405 shows the fields that are included in the template. The fields are shown in the order in which they appear in the template.

Table 60: IPFIX MPLS-IPv4 Template Fields for MX, M, and T Series

Field	Element ID
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72
MPLS Top Label IP Address	47
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 ToS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
VLAN ID	58
IPv4 Source Mask	9
IPv4 Destination Mask	13
Source AS	16
Destination AS	17
IPv4 Next Hop Address	15

Table 60: IPFIX MPLS-IPv4 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
TCP Flags	6
Output Interface	14
Minimum TTL	52
Maximum TTL	53
Flow End Reason	136
IP Protocol Version	60
BGP IPv4 Next Hop Address	18
Flow Direction	61
802.1Q VLAN identifier (dot1qVlanId)	243
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX MPLS-IPv6 Template for MX, M, and T Series

Starting in Junos OS Release 18.4R1, the IPFIX MPLS-IPv6 template is supported for the MX Series. Table 61 on page 406 shows the fields that are included in the template.

Table 61: IPFIX MPLS-IPv6 Template Fields for MX, M, and T Series

Field	Element ID
MPLS Label 1	70
MPLS Label 2	71

Table 61: IPFIX MPLS-IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
MPLS Label 3	72
MPLS Top Label IP Address (Only IPv4 top label addresses are exported. IPv6 top label addresses report a value of zero.)	47
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 ToS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	139
Input Interface	10
VLAN ID	58
IPv6 Source Mask	29
IPv6 Destination Mask	30
Source AS	16
Destination AS	17
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
TCP Flags	6
Output Interface	14
Minimum TTL	52

Table 61: IPFIX MPLS-IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Maximum TTL	53
Flow End Reason	136
Flow Direction	61
802.1Q VLAN identifier (dot1qVlanId)	243
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
IPv6 Option Headers	64
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX MPLS-IPv4 Template for PTX Series

Starting in Junos OS Release 18.2R1, the IPFIX MPLS-IPv4 template is supported for the PTX Series. Table 62 on page 408 shows the fields that are included in the template.

Table 62: IPFIX MPLS-IPv4 Template Fields for PTX Series

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 ToS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11

Table 62: IPFIX MPLS-IPv4 Template Fields for PTX Series *(continued)*

Field	Element ID
ICMP Type and Code	32
Input Interface	10
Source AS	16
Destination AS	17
BGP IPv4 Next Hop Address	18
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153
IPv4 Next Hop Address	15
IPv4 Source Mask	9
IPv4 Destination Mask	13
TCP Flags	6
IP Protocol Version	60
Ingress Interface Type	368
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72

Table 62: IPFIX MPLS-IPv4 Template Fields for PTX Series *(continued)*

Field	Element ID
MPLS Top Label IPv6 Address	140

Fields Included in the IPFIX MPLS-IPv6 Template for PTX Series

Starting in Junos OS Release 18.2R1, the IPFIX MPLS-IPv6 template is supported for the PTX Series. Table 63 on page 410 shows the fields that are included in the IPFIX MPLS-IPv6 template.

Table 63: IPFIX MPLS-IPv6 Template Fields for PTX Series

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 ToS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	139
Input Interface	10
Source AS	16
Destination AS	17
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152

Table 63: IPFIX MPLS-IPv6 Template Fields for PTX Series *(continued)*

Field	Element ID
Time the flow ended with respect to Epoch time	153
IPv6 Source Mask	29
IPv6 Destination Mask	30
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
TCP Flags	6
IP protocol version of IP payload on MPLS VPN	60
Ingress Interface Type	368
RSVP label (top MPLS label stack entry) for MPLS tunnel	70
RSVP label pushed before top label	71
MPLS Label 3	72
MPLS Top Label IPv6 Address	140

Fields Included in the IPFIX MPLS Template for MX, M, and T Series

Starting in Junos OS Release 16.1, the IPFIX MPLS template is supported. Table 64 on page 411 shows the fields that are included in the template. The fields are shown in the order in which they appear in the template.

Table 64: IPFIX MPLS Template Fields for MX, M, and T Series

Field	Element ID
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72
Input Interface	10

Table 64: IPFIX MPLS Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Output Interface	14
Flow End Reason	136
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX MPLS Template for PTX Series

Starting in Junos OS Release 18.2R1, the IPFIX MPLS template is supported for the PTX Series. Table 65 on page 412 shows the fields that are included in the template.

Table 65: IPFIX MPLS Template Fields for PTX Series

Field	Element ID
Input Interface	10
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153
Ingress Interface Type	368
MPLS Label 1	70
MPLS Label 2	71

Table 65: IPFIX MPLS Template Fields for PTX Series *(continued)*

Field	Element ID
MPLS Label 3	72

Fields Included in the IPFIX MPLS-over-UDP Template for PTX Series for Flows Within an IP Network and Having an IPv4 Payload

Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series. Table 66 on page 413 shows the fields that are available in the IPFIX template for MPLS-over-UDP flows that are within an IP network and have an IPv4 payload.

Table 66: IPFIX MPLS-over-UDP Carried on IP Network Template Fields (IPv4 Payload) for PTX Series

Field	Element ID
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP IPv4 next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv4 source address of tunnel payload	8

Table 66: IPFIX MPLS-over-UDP Carried on IP Network Template Fields (IPv4 Payload) for PTX Series *(continued)*

Field	Element ID
IPv4 destination address of tunnel payload	12
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11
ICMP type	32
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX MPLS-over-UDP Template for PTX Series for Flows Encapsulated in an RSVP-TE LSP and Having an IPv4 Payload

Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series. Table 67 on page 414 shows the fields that are available in the IPFIX template for MPLS-over-UDP flows that are encapsulated in an RSVP-TE LSP in the inner MPLS network and have an IPv4 payload.

Table 67: IPFIX MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv4 Payload) for PTX Series

Field	Element ID
RSVP label (top MPLS label stack entry) for MPLS tunnel	70
RSVP label pushed before top label	71

Table 67: IPFIX MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv4 Payload) for PTX Series *(continued)*

Field	Element ID
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP IPv4 next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv4 source address of tunnel payload	8
IPv4 destination address of tunnel payload	12
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11

Table 67: IPFIX MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv4 Payload) for PTX Series *(continued)*

Field	Element ID
ICMP type	32
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX MPLS-over-UDP Template for PTX Series for Flows Within an IP Network Having an IPv6 Payload

Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series. Table 68 on page 416 shows the fields that are available in the IPFIX template for MPLS-over-UDP flows that are within an IP network and have an IPv6 payload.

Table 68: IPFIX MPLS-over-UDP Carried on IP Network Template Fields (IPv6 Payload) for PTX Series

Field	Element ID
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17

Table 68: IPFIX MPLS-over-UDP Carried on IP Network Template Fields (IPv6 Payload) for PTX Series *(continued)*

Field	Element ID
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv6 source address of tunnel payload	27
IPv6 destination address of tunnel payload	28
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11
ICMP type V6	139
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX MPLS-over-UDP Template for PTX Series for Flows Encapsulated in an RSVP-TE LSP and Having an IPv6 Payload

Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series. Table 69 on page 418 shows the fields that are available in the IPFIX template for MPLS-over-UDP flows that are encapsulated in an RSVP-TE LSP in the inner MPLS network and have an IPv6 payload.

Table 69: IPFIX MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv6 Payload) for PTX Series

Field	Element ID
RSVP label (top MPLS label stack entry) for MPLS tunnel	70
RSVP label pushed before top label	71
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv6 source address of tunnel payload	27

Table 69: IPFIX MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv6 Payload) for PTX Series *(continued)*

Field	Element ID
IPv6 destination address of tunnel payload	28
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11
ICMP type V6	139
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the IPFIX VPLS Template for MX, M, and T Series

Starting in Junos OS Release 16.1, the IPFIX VPLS template is supported. Table 70 on page 419 shows the fields that are included in the template. The fields are shown in the order in which they appear in the template.

Table 70: IPFIX VPLS Template Fields for MX, M, and T Series

Field	Element ID
Destination MAC	80
Source MAC	56
Ethernet Type	256

Table 70: IPFIX VPLS Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Input Interface	10
Output Interface	14
Flow End Reason	136
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Fields Included in the Version 9 Bridge Template for MX Series

Table 71 on page 420 shows the fields that are included in the version 9Bridge template. The fields are shown in the order in which they appear in the template.

Table 71: Version 9 Bridge Template Fields for MX

Field	Element ID
Destination MAC	80
Source MAC	56
Ethernet Type	256
Input SNMP	10
Output SNMP	14
Flow End Reason	136
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to Epoch time	22
Time the flow ended with respect to Epoch time	21

Fields Included in the Version 9 IPv4 Template for MX, M, and T Series

Table 72 on page 421 shows the fields that are included in the version 9 IPv4 template. The fields are shown in the order in which they appear in the template.

Table 72: Version 9 IPv4 Template Fields for MX, M, and T Series

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 ToS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
VLAN ID	58
IPv4 Source Mask	9
IPv4 Destination Mask	13
Source AS	16
Destination AS	17
IPv4 Next Hop Address	15
TCP Flags	6
Output Interface	14
Minimum TTL	52
Maximum TTL	53
Flow End Reason	136

Table 72: Version 9 IPv4 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Internet Protocol Version	60
BGP IPv4 Next Hop Address	18
Flow Direction	61
802.1Q VLAN identifier (dot1qVlanId)	243
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 IPv4 Template for PTX Series

Table 73 on page 422 shows the fields that are available in the IPv4 templates.

Table 73: Version 9 IPv4 Template Fields for PTX Series

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 TOS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32

Table 73: Version 9 IPv4 Template Fields for PTX Series *(continued)*

Field	Element ID
Input Interface	10
Source AS	16
Destination AS	17
BGP IPv4 Next Hop Address	18
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv4 Next Hop Address	15
IPv4 Source Mask	9
IPv4 Destination Mask	13
TCP Flags	6
IP Protocol Version	60

Fields Included in the Version 9 IPv4 Template for PTX10003-160C and PTX10003-80C routers

Table 74 on page 423 shows the fields that are available in the IPv4 templates.

Table 74: Version 9 IPv4 Template Fields for PTX10003-160C and PTX10003-80C routers

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 TOS	5

Table 74: Version 9 IPv4 Template Fields for PTX10003-160C and PTX10003-80C routers *(continued)*

Field	Element ID
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input SNMP Index	10
IPv4 Source Mask	9
IPv4 Destination Mask	13
Source AS	16
Destination AS	17
IPv4 Next Hop Address	15
BGP IPv4 Next Hop Address	18
TCP Flags	6
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IP Protocol Version	60
Output SNMP Index	14

Fields Included in the Version 9 IPv6 Template for MX, M, and T Series

Starting in Junos OS Release 18.1R1, the version 9 IPv6 template is supported. Table 75 on page 425 shows the fields in the template. The fields are shown in the order in which they appear in the template.

Table 75: Version 9 IPv6 Template Fields for MX, M, and T Series

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 ToS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	139
Input Interface	10
VLAN ID	58
IPv6 Source Mask	29
IPv6 Destination Mask	30
Source AS	16
Destination AS	17
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
TCP Flags	6
Output Interface	14
Minimum TTL	52
Maximum TTL	53
Flow End Reason	136
Flow Direction	61

Table 75: Version 9 IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
802.1Q VLAN identifier (dot1qVlanId)	243
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
IPv6 Option Headers	64
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 IPv6 Template for PTX Series

Table 76 on page 426 shows the fields that are available in the IPv6 templates.

Table 76: IPv6 Template Fields for PTX Series

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 TOS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
Source AS	16

Table 76: IPv6 Template Fields for PTX Series *(continued)*

Field	Element ID
Destination AS	17
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
IPv6 Source Mask	29
IPv6 Destination Mask	30
TCP Flags	6
IP Protocol Version	60

Fields Included in the Version 9 IPv6 Template for PTX10003-160C and PTX10003-80C routers

Table 77 on page 427 shows the fields that are available in the IPv6 templates.

Table 77: IPv6 Template Fields for PTX10003-160C and PTX10003-80C routers

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 TOS	5
IPv6 Protocol	4
L4 Source Port	7

Table 77: IPv6 Template Fields for PTX10003-160C and PTX10003-80C routers *(continued)*

Field	Element ID
L4 Destination Port	11
ICMP Type and Code	32
Input SNMP Index	10
Output SNMP Index	14
IPv6 Source Mask	29
IPv6 DestinationMask	30
Source AS	16
Destination AS	17
IPv6 Next Hop Address	62
TCP Flags	6
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IP Protocol Version	60

Fields Included in the Version 9 MPLS-IPv4 Template for MX, M, and T Series

Starting in Junos OS Release 16.1, the version 9 MPLS-IPv4 template is supported. Table 78 on page 428 shows the fields that are included in the template. The fields are shown in the order in which they appear in the template.

Table 78: Version 9 MPLS-IPv4 Template Fields for MX, M, and T Series

Field	Element ID
MPLS Label 1	70

Table 78: Version 9 MPLS-IPv4 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
MPLS Label 2	71
MPLS Label 3	72
MPLS Top Label IP Address	47
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 ToS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
VLAN ID	58
IPv4 Source Mask	9
IPv4 Destination Mask	13
Source AS	16
Destination AS	17
IPv4 Next Hop Address	15
TCP Flags	6
Output Interface	14
Minimum TTL	52
Maximum TTL	53

Table 78: Version 9 MPLS-IPv4 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
Flow End Reason	136
IP Protocol Version	60
BGP IPv4 Next Hop Address	18
Flow Direction	61
802.1Q VLAN identifier (dot1qVlanId)	243
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 MPLS-IPv6 Template for MX, M, and T Series

Starting in Junos OS Release 18.4R1, the version 9 MPLS-IPv6 template is supported for the MX Series. Table 79 on page 430 shows the fields that are included in the template.

Table 79: Version 9 MPLS-IPv6 Template Fields for MX, M, and T Series

Field	Element ID
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72
MPLS Top Label IP Address (Only IPv4 top label addresses are exported. IPv6 top label addresses report a value of zero.)	47
IPv6 Source Address	27

Table 79: Version 9 MPLS-IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
IPv6 Destination Address	28
IPv6 ToS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	139
Input Interface	10
VLAN ID	58
IPv6 Source Mask	29
IPv6 Destination Mask	30
Source AS	16
Destination AS	17
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
TCP Flags	6
Output Interface	14
Minimum TTL	52
Maximum TTL	53
Flow End Reason	136
Flow Direction	61
802.1Q VLAN identifier (dot1qVlanId)	243

Table 79: Version 9 MPLS-IPv6 Template Fields for MX, M, and T Series *(continued)*

Field	Element ID
802.1Q Customer VLAN identifier (dot1qCustomerVlanId)	245
IP Identifier	54
IPv6 Option Headers	64
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 MPLS-IPv4 Template for PTX Series

Starting in Junos OS Release 18.2R1, the version 9 MPLS-IPv4 template is supported for the PTX Series. Table 80 on page 432 shows the fields that are included in the template.

Table 80: Version 9 MPLS-IPv4 Template Fields for PTX Series

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 ToS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
Source AS	16
Destination AS	17

Table 80: Version 9 MPLS-IPv4 Template Fields for PTX Series *(continued)*

Field	Element ID
BGP IPv4 Next Hop Address	18
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv4 Next Hop Address	15
IPv4 Source Mask	9
IPv4 Destination Mask	13
TCP Flags	6
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72
MPLS Top Label IP Address	47

Fields Included in the Version 9 MPLS-IPv6 Template for PTX Series

Starting in Junos OS Release 18.2R1, the version 9 MPLS-IPv6 template is supported for the PTX Series. Table 81 on page 433 shows the fields that are included in the version 9 MPLS-IPv6 template.

Table 81: Version 9 MPLS-IPv6 Template Fields for PTX Series

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 ToS	5

Table 81: Version 9 MPLS-IPv6 Template Fields for PTX Series *(continued)*

Field	Element ID
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	32
Input Interface	10
Source AS	16
Destination AS	17
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv6 Source Mask	29
IPv6 Destination Mask	30
IPv6 Next Hop Address	62
IPv6 BGP NextHop Address	63
TCP Flags	6
IP protocol version of IP payload on MPLS VPN	60
RSVP label (top MPLS label stack entry) for MPLS tunnel	70
RSVP label pushed before top label	71
MPLS Label 3	72

Table 81: Version 9 MPLS-IPv6 Template Fields for PTX Series *(continued)*

Field	Element ID
MPLS Top Label IP Address	47

Fields Included in the Version 9 MPLS Template for MX, M, and T Series

Starting in Junos OS Release 16.1, the version 9 MPLS template is supported. Table 82 on page 435 shows the fields that are included in the template. The fields are shown in the order in which they appear in the template.

Table 82: Version 9 MPLS Template Fields for MX, M, and T Series

Field	Element ID
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72
Input Interface	10
Output Interface	14
Flow End Reason	136
Number of Flow Bytes	1
Number of Flow Packets	2
First Switched	ww
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 MPLS Template for PTX Series

Starting in Junos OS Release 18.2R1, the version 9 MPLS template is supported for the PTX Series. Table 83 on page 435 shows the fields that are included in the template.

Table 83: Version 9 MPLS Template Fields for PTX Series

Field	Element ID
Input Interface	10
Output Interface	14

Table 83: Version 9 MPLS Template Fields for PTX Series *(continued)*

Field	Element ID
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
MPLS Label 1	70
MPLS Label 2	71
MPLS Label 3	72

Fields Included in the Version 9 MPLS-over-UDP Template for PTX Series for Flows Within an IP Network Having an IPv4 Payload

Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series. Table 84 on page 436 shows the fields that are available in the Version 9 template for MPLS-over-UDP flows that are within an IP network and have an IPv4 payload.

Table 84: Version 9 MPLS-over-UDP Carried on IP Network Template Fields (IPv4 Payload) for PTX Series

Field	Element ID
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17
IPv4 next hop address—gateway for tunnel destination IP address	15

Table 84: Version 9 MPLS-over-UDP Carried on IP Network Template Fields (IPv4 Payload) for PTX Series *(continued)*

Field	Element ID
BGP IPv4 next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv4 source address of tunnel payload	8
IPv4 destination address of tunnel payload	12
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11
ICMP type	32
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 MPLS-over-UDP Template for PTX Series for Flows Encapsulated in an RSVP-TE LSP and Having an IPv4 Payload

Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series. Table 85 on page 438 shows the fields that are available in the Version 9 template for MPLS-over-UDP flows that are encapsulated in an RSVP-TE LSP in the inner MPLS network and have an IPv4 payload.

Table 85: Version 9 MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv4 Payload) for PTX Series

Field	Element ID
RSVP label (top MPLS label stack entry) for MPLS tunnel	70
RSVP label pushed before top label	71
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP IPv4 next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv4 source address of tunnel payload	8
IPv4 destination address of tunnel payload	12
IP protocol of tunnel payload	4
IP TOS	5

Table 85: Version 9 MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv4 Payload) for PTX Series *(continued)*

Field	Element ID
Source transport port	7
Destination transport port	11
ICMP type	32
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 MPLS-over-UDP Template for PTX Series for Flows Within an IP Network Having an IPv6 Payload

Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series. Table 86 on page 439 shows the fields that are available in the Version 9 template for MPLS-over-UDP flows that are within an IP network and have an IPv6 payload.

Table 86: Version 9 MPLS-over-UDP Carried on IP Network Template Fields (IPv6 Payload) for PTX Series

Field	Element ID
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17

Table 86: Version 9 MPLS-over-UDP Carried on IP Network Template Fields (IPv6 Payload) for PTX Series *(continued)*

Field	Element ID
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv6 source address of tunnel payload	27
IPv6 destination address of tunnel payload	28
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11
ICMP type V6	32
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Fields Included in the Version 9 MPLS-over-UDP Template for PTX Series for Flows Encapsulated in an RSVP-TE LSP and Having an IPv6 Payload

Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series. Table 87 on page 441 shows the fields that are available in the Version 9 template for MPLS-over-UDP flows that are encapsulated in an RSVP-TE LSP in the inner MPLS network and have an IPv6 payload.

Table 87: Version 9 MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv6 Payload) for PTX Series

Field	Element ID
RSVP label (top MPLS label stack entry) for MPLS tunnel	70
RSVP label pushed before top label	71
IPv4 source address for tunnel endpoint	8
IPv4 destination address for tunnel endpoint	12
UDP source port for tunnel endpoint	7
Tunnel endpoint destination transport port	11
IPv4 source mask for tunnel source IP address	9
IPv4 destination mask for tunnel destination IP address	13
Source AS for tunnel	16
Destination AS for tunnel	17
IPv4 next hop address—gateway for tunnel destination IP address	15
BGP next hop address—tunnel destination IP BGP peer	18
Input SNMP index	10
Output SNMP index	14
MPLS label 1—VPN bottom of stack label	70
IP protocol version of IP payload on MPLS VPN	60
IPv6 source address of tunnel payload	27

Table 87: Version 9 MPLS-over-UDP Encapsulated in RSVP-TE LSP Template Fields (IPv6 Payload) for PTX Series *(continued)*

Field	Element ID
IPv6 destination address of tunnel payload	28
IP protocol of tunnel payload	4
IP TOS	5
Source transport port	7
Destination transport port	11
ICMP type	32
TCP flags	6
Number of flow bytes	1
Number of flow packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21

Release History Table

Release	Description
19.4R1	Starting with Junos OS Release 19.4R1 on the PTX10002-60C router, you can perform flow monitoring for MPLS-over-UDP flows to look past the tunnel header to sample and report on the inner payload at both the transit and egress nodes of the tunnel. MPLS IPv4 and IPv6 payloads and IPFIX and version 9 templates are supported. Only ingress sampling is supported.
19.3R2	Inline active flow monitoring is not supported when you enable Next Gen Services on an MX Series router.
18.4R1	Starting in Junos OS Release 18.4R1, you can configure inline active flow monitoring for MPLS-IPv6 traffic for MX Series routers.
18.4R1	Starting in Junos OS Release 18.4R1, the IPFIX MPLS-IPv6 template is supported for the MX Series.
18.4R1	Starting in Junos OS Release 18.4R1, the version 9 MPLS-IPv6 template is supported for the MX Series.
18.2R1	Starting in Junos OS Release 18.2R1, you can configure inline active flow monitoring for MPLS, MPLS-IPv4, and MPLS-IPv6 traffic for PTX Series routers.
18.2R1	Starting in Junos OS Release 18.2R1, you can configure inline active flow monitoring for bridge traffic for MX Series routers.
18.2R1	Starting in Junos OS Release 18.2R1, the IPFIX MPLS-IPv4 template is supported for the PTX Series.
18.2R1	Starting in Junos OS Release 18.2R1, the IPFIX MPLS-IPv6 template is supported for the PTX Series.
18.2R1	Starting in Junos OS Release 18.2R1, the IPFIX MPLS template is supported for the PTX Series.
18.2R1	Starting in Junos OS Release 18.2R1, the version 9 MPLS-IPv4 template is supported for the PTX Series.
18.2R1	Starting in Junos OS Release 18.2R1, the version 9 MPLS-IPv6 template is supported for the PTX Series.
18.2R1	Starting in Junos OS Release 18.2R1, the version 9 MPLS template is supported for the PTX Series.

18.1R1	Starting in Junos OS Release 18.1R1, you can configure inline active flow monitoring for MPLS-over-UDP traffic for PTX Series routers, except for the PTX10002-60C.
18.1R1	Starting in Junos OS Release 18.1R1, you can also apply version 9 flow templates to IPv6 traffic.
18.1R1	Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series.
18.1R1	Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series.
18.1R1	Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series.
18.1R1	Starting in Junos OS Release 18.1R1, the IPFIX MPLS-Over-UDP template is supported for the PTX Series.
17.4R1	Starting with Junos OS Release 17.4R1, you can apply IPFIX flow templates to unicast IPv4 and IPv6 traffic on QFX10008 and QFX10016 switches.
17.2R1	Starting with Junos OS Release 17.2R1, you can apply IPFIX flow templates to unicast IPv4 and IPv6 traffic on QFX10002 switches.
16.1R4	In Junos OS Release 16.2 and in Junos OS Release 16.1R3 and earlier, you can configure only one collector under a family for inline active flow monitoring. Starting with Junos OS Release 16.1R4 and 17.2R1, you can configure up to four collectors under a family for inline active flow monitoring.
16.1R1	Starting with Junos OS Release 16.1R1 and 15.1F2, the IPv4 table is assigned a default value of 1024.
16.1	Starting in Junos OS Release 16.1, you can also apply version 9 flow templates to MPLS and MPLS-IPv4 traffic.
16.1	Starting in Junos OS release 16.1, you can also apply IPFIX flow templates to MPLS and MPLS-IPv4 traffic.
16.1	Flow Direction (Starting in Junos OS Release 16.1)
16.1	Flow Direction (Starting in Junos OS Release 16.1)
16.1	Starting in Junos OS Release 16.1, the IPFIX MPLS-IPv4 template is supported.

16.1	Starting in Junos OS Release 16.1, the IPFIX MPLS template is supported.
16.1	Starting in Junos OS Release 16.1, the IPFIX VPLS template is supported.
16.1	Starting in Junos OS Release 18.1R1, the version 9 IPv6 template is supported.
16.1	Starting in Junos OS Release 16.1, the version 9 MPLS-IPv4 template is supported.
16.1	Starting in Junos OS Release 16.1, the version 9 MPLS template is supported.
16.1	Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series.
16.1	Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series.
16.1	Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series.
16.1	Starting in Junos OS Release 18.1R1, the version 9 MPLS-over-UDP template is supported for the PTX Series.

RELATED DOCUMENTATION

Example: Configuring Inline Active Flow Monitoring on MX Series and T4000 Routers | 479

Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers | 455

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250

Active flow monitoring is implemented on the Packet Forwarding Engine. The Packet Forwarding Engine performs functions such as creating and updating flows, and updating flow records. The flow records are sent out in industry-standard IPFIX or version 9 format. Support for active flow monitoring with IPFIX templates on QFX10002 switches was added in Junos OS Release 17.2R1.

On routers with MS-PICs or MS-DPCs, IPv4 and IPv6 fragments are processed accurately. The flow monitoring application creates two flows for every fragmented flow. The first fragment that has the

complete Layer 4 information forms the first flow with 5-tuple data and subsequently, all the fragmented packets related to this flow form another flow with the Layer 4 fields set to zero.

The following considerations apply to the inline flow-monitoring instance configuration:

- Sampling run-length and clip-size are not supported.
- For inline configurations, collectors are not reachable via **fxp0**.
- Inline flow monitoring does not support **cflowd**. Therefore, inline flow monitoring does not support the local dump option, which is available only with cflowd.
- Inline active flow monitoring is not supported when you enable Next Gen Services on an MX Series router.
- The number of collectors that are supported depends on the device:
  - On MX Series routers running Junos OS Release 16.1R4 and later, you can export flow records to four collectors under a family with the same source IP address for Inline-JFlow. The Packet Forwarding Engine (PFE) can export the flow record, flow record template, option data, and option data template packet to all configured collectors. You can configure the multiple collectors at the **[edit forwarding-options sampling instance instance name]** hierarchy level.
  - For inline configurations on all other devices, each family can support only one collector.

Inline active flow monitoring is available in four hierarchies levels:

- **[edit chassis]** —At this level, you associate the sampling instance with the FPC on which the media interface is present (except on the MX80 and MX104—see "Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers" on page 455). If you are configuring sampling of IPv4 flows, IPv6 flows or VPLS flows, you can configure the flow hash table size for each family, as described below.
- **[edit firewall]**—At this level, you configure a firewall filter for the family of traffic to be sampled. You must attach this filter to the interface on which you want to sample the traffic.
- **[edit forwarding-options]**—At this level, you configure a sampling instance and associate the template with the sampling instance. At this level, you also configure the flow-server IP address and port number as well as the flow export rate.
- **[edit services flow-monitoring]** —At this level, you configure the template properties for inline flow monitoring.

Before you configure inline active flow monitoring, you should ensure that you have adequately-sized hash tables for IPv4, IPv6, MPLS, and VPLS flow sampling. These tables can use one to fifteen 256K areas. Starting with Junos OS Release 16.1R1 and 15.1F2, the IPv4 table is assigned a default value of 1024. Prior to Junos OS Release 16.1 and 15.1F2, the IPv4 table is assigned a default value of fifteen 256K areas. The IPv6 table is assigned a default value of 1024, and the VPLS table is assigned a default value of 1024. When anticipated traffic volume requires larger tables, allocate larger tables.

To allocate flow hash tables:

1. Go to the [edit-flow-table-size] hierarchy level for inline services on the FPC that processes the monitored flows.

```
[edit]
user@host# edit chassis fpc 0 inline-services flow-table-size
```

2. Specify the required sizes for the sampling hash tables.

```
[edit chassis fpc 0 inline-services flow-table-size]
user@host# set bridge-flow-table-sizeunits
user@host# set ipv4-flow-table-size units
user@host# set ipv6-flow-table-size units
user@host# set mpls-flow-table-size units
user@host#set vpls-flow-table-size units
```

> **NOTE:** Starting in Junos OS Release 18.2R1, the **bridge-flow-table-size** option is available and the **vpls-flow-table-size** option is deprecated; use the **bridge-flow-table-size** option instead. The **bridge-flow-table-size** option supports both VPLS and bridge records.

> **NOTE:** The total number of units used for IPv4, IPv6, MPLS, and VPLS cannot exceed 15. Also, starting in Junos OS Release 16.1R1 and 15.1F2, changing the flow hash table size does *not* automatically reboot the FPC (for earlier releases changing the flow hash table size triggers the FPC to reboot).

To configure inline active flow monitoring on MX Series routers (except for MX80 and MX104 routers), EX Series switches, and T4000 routers with Type 5 FPC:

1. Enable inline active flow monitoring and specify the source address for the traffic.

```
[edit forwarding-options sampling instance instance-name family (bridge | inet | inet6 | mpls | vpls ) output]
user@host# set inline-jflow source address address
```

2. Specify the template to use with the sampling instance.

```
[edit forwarding-options sampling instance instance-name family (bridge | inet | inet6 | mpls | vpls ) output
  flow-server hostname]
user@host# set (version9 | version-ipfix) template template-name
```

3. Configure a template to specify output properties.

```
[edit services flow-monitoring]
user@host# set (version-ipfix | version9) template template-name
```

4. (Optional) Configure the interval after which an active flow is exported.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-active-timeout seconds
```

5. (Optional) Configure the interval of activity that marks a flow as inactive.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-inactive-timeout seconds
```

6. (Optional) Configure the template refresh rate in either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set template-refresh-rate packets packets seconds seconds
```

7. (Optional) Configure the refresh rate in either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set option-refresh-rate packets packets seconds seconds
```

8. Specify the type of record that the template is used for.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set (bridge-templateipv4-template |ipv6-template | mpls-ipv4-template | mpls-template |
  peer-as-billing-template | vpls-template)
```

The **vpls-template** option is only for IPFIX templates.

Starting in Junos OS Release 18.2R1, the **bridge-template** option is available and the **vpls-template** option is deprecated; use the **bridge-template** option instead. The **bridge-template** option supports both VPLS and bridge records and is for both IPFIX and version9 templates.

Starting in Junos OS Release 18.4R1, the **MPLS-ipv4-template** option is deprecated for inline flow monitoring. To configure MPLS records starting in Junos OS Release 18.4R1, use the **mpls-template** option and the **tunnel-observation** option. This is described in step 9.

9. Starting in Junos OS Release 18.4R1 for the MX Series, if you are configuring any type of MPLS flow records, perform the following:

   a. Specify the MPLS template.

   ```
   [edit services flow-monitoring (version-ipfix | version9) template template-name]
   user@host# set mpls-template
   ```

   b. Configure the type of MPLS flow records to create.

   ```
   [edit services flow-monitoring (version-ipfix | version9) template template-name]
   user@host# set tunnel-observation [ipv4 | ipv6]
   ```

   The **tunnel-observation** values enable the creation of the following types of flow records:

   - **ipv4**—MPLS-IPv4 flows
   - **ipv6**—MPLS-IPv6 flows

   You can configure multiple values for **tunnel-observation**.

   For an MPLS traffic type that does *not* match any of the **tunnel-observation** values, plain MPLS flow records are created. For example, if you only configure **ipv4**, then MPLS-IPv6 traffic results in plain MPLS flow records.

   If you do not configure **tunnel-observation**, plain MPLS flow records are created.

   c. If you are running inline flow monitoring on a Lookup (LU) card, enable sideband mode to create MPLS-IPv6 flow records.

   ```
   [edit chassis fpc slot-number inline-services]
   user@host# set use-extended-flow-memory
   ```

   If you are running inline flow monitoring on an LU card and do not enable sideband mode, then MPLS-IPv6 traffic results in plain MPLS flow records.

10. (Optional) Include the flow direction value in the template.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-key flow-direction
```

The reported data field contains 0x00 (ingress) or 0x01 (egress). If you do not include the **flow-key flow-direction** statement, the flow direction data field contains the invalid value 0xFF.

11. (Optional) Include VLAN IDs in both the ingress and egress directions in the flow key.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-key vlan-id
```

This statement is not required for ingress and egress VLAN ID reporting on interfaces.

12. Associate the sampling instance with the FPC on which you want to implement inline active flow monitoring.

For MX240, MX480, MX960, MX2010, MX2020, use the following command:

```
[edit ]
user@host# set chassis fpc fpc-number sampling-instance instance-name.
```

a. Confirm the configuration by running the following show command:

```
user@host# show chassis
```

```
fpc 0 {
        sampling-instance sample-ins1;
    }
```

For MX5, MX10, MX40, and MX80, use the following command:

```
[edit ]
user@host# set chassis tfeb slot 0 sampling-instance instance-name.
```

a. Confirm the configuration by running the following show command:

```
user@host# show chassis
```

```
tfeb {
    slot 0 {
        sampling-instance sample-ins1;
    }
}
```

For MX104, use the following command:

```
[edit ]
user@host# set chassis afeb slot 0 sampling-instance instance-name.
```

a. Confirm the configuration by running the following show command:

```
user@host# show chassis
```

```
afeb {
    slot 0 {
        sampling-instance sample-ins1;
    }
}
```

This example shows the sampling configuration for an instance that supports inline active flow monitoring on **family inet**:

```
[edit]
user@host> show forwarding-options
sampling {
   instance {
      sample-ins1 {
         input {
            rate 1;
         }
         family inet {
            output {
               flow-server 192.0.2.2 {
                  port 2055;
                  version-ipfix {
                     template {
                        ipv4;
                     }
                  }
```

```
            }
            inline-jflow {
                source-address 10.11.12.13;
            }
        }
    }
  }
}
```

Here is the output format configuration:

```
[edit]
user@host> show services flow-monitoring
services {
   flow-monitoring {
      version-ipfix {
         template ipv4 {
            flow-active-timeout 60;
            flow-inactive-timeout 60;
            ipv4-template;
            template-refresh-rate {
               packets 1000;
               seconds 10;
            }
            option-refresh-rate {
               packets 1000;
               seconds 10;
            }
         }
      }
   }
}
```

The following example shows the output format configuration for chassis **fpc** 0:

```
[edit]
user@host> show services flow-monitoring
sampling-instance instance-1; {
   inline-services {
      flow-table-size {
         ipv4-flow-table-size 8;
         ipv6-flow-table-size 7;
```

```
        }
      }
    }
```

Release History Table

Release	Description
19.3R2	Inline active flow monitoring is not supported when you enable Next Gen Services on an MX Series router.
18.4R1	Starting in Junos OS Release 18.4R1, the **MPLS-ipv4-template** option is deprecated for inline flow monitoring. To configure MPLS records starting in Junos OS Release 18.4R1, use the **mpls-template** option and the **tunnel-observation** option.
18.2R1	Starting in Junos OS Release 18.2R1, the **bridge-flow-table-size** option is available and the **vpls-flow-table-size** option is deprecated; use the **bridge-flow-table-size** option instead.
18.2R1	Starting in Junos OS Release 18.2R1, the **bridge-template** option is available and the **vpls-template** option is deprecated; use the **bridge-template** option instead.
16.1R4	On MX Series routers running Junos OS Release 16.1R4 and later, you can export flow records to four collectors under a family with the same source IP address for Inline-JFlow.
16.1R1	Starting with Junos OS Release 16.1R1 and 15.1F2, the IPv4 table is assigned a default value of 1024.
16.1R1	Also, starting in Junos OS Release 16.1R1 and 15.1F2, changing the flow hash table size does *not* automatically reboot the FPC

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers | **455**

Example: Configuring Inline Active Flow Monitoring on MX Series and T4000 Routers | **479**

inline-jflow | **991**

Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers

To configure inline active flow monitoring on MX80 and MX104 routers:

1. Associate a sampling instance with the Forwarding Engine Processor.

   For the MX80:

   ```
   [edit]
   user@host# set chassis tfeb slot 0 sampling-instance sampling-instance
   ```

   The Forwarding Engine Processor slot is always **0** because MX80 and MX104 routers have only one Packet Forwarding Engine. In this MX80 configuration, the sampling instance is **sample-ins1**.

   ```
   [edit]
   user@host# set chassis tfeb slot 0 sampling-instance sample-ins1
   ```

   For the MX104:

   ```
   [edit]
   user@host# set chassis afeb slot 0 sampling-instance sampling-instance
   ```

   NOTE: MX80 and MX104 routers support only one sampling instance.

2. Under forwarding-options, configure a sampling instance for the flow server and inline jflow instances (these be configured in the following steps):

   ```
   [edit forwarding-options sampling]
   user@host# edit instance inline_sample
   ```

3. Configure the rate at the **[edit forwarding-options sampling instance instance-name input]** hierarchy level to apply specific values for the sampling instance **sample-ins1**.

   ```
   [edit forwarding-options sampling instance sample-ins1 input]
   user@host# set rate number
   ```

In this configuration, the rate is **1000**.

```
[edit forwarding-options sampling instance sample-ins1 input]
user@host# set rate 1000
```

4. Navigate to the output hierarchy and from there, enable a flow server and then specify the output address and port:

```
[edit] forwarding-options sampling instance inline_sample family inet output]
user@host# edit flow-server hostname
```

```
[edit forwarding-options sampling instance inline_sample family inet output flow-server hostname]
user@host# set port number
```

5. Return to the output hierarchy and specify the source address for inline jflow:

```
[edit forwarding-options sampling instance sample-ins1 family inet output]
user@host# set inline-jflow source-address address
```

In this configuration, the source address is **10.11.12.13**.

```
[edit forwarding-options sampling instance sample-ins1 family inet output]
user@host# set inline-jflow source-address 10.11.12.13
```

6. Specify the output properties.

```
[edit services flow-monitoring]
user@host# set version-ipfix
```

The output format properties are common to other output formats and are described in "“Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices” on page 513".

The following is an example of the sampling configuration for an instance that supports inline active flow monitoring on MX80 routers:

```
[edit forwarding-options]
user@host# show
sampling {
```

```
    instance {
      inline_sample {
        input {
          rate 1000;
        }
        family inet{
          output {
            flow-server 192.168.64.143 {
              port 80;
            }
            inline-jflow {
              source-address 10.10.11.12;
            }
          }
        }
      }
    }
}
```

NOTE: You need not configure a Flexible PIC Concentrator (FPC) slot because MX80 routers have only one Packet Forwarding Engine.

The following considerations apply to the inline flow-monitoring instance configuration:

- This configuration does not support MPLS-IPv6.
- Clip-size is not supported.

RELATED DOCUMENTATION

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | **63**

inline-jflow | **991**

Configuring Inline Active Flow Monitoring on PTX Series Routers

IN THIS SECTION

This topic describes how to configure inline flow monitoring on PTX Series routers for IPv4 and IPv6 traffic. Starting in Junos OS Release 18.2R1, you can also configure inline flow monitoring on the PTX Series for MPLS, MPLS-IPv4, and MPLS-IPv6 traffic. To configure inline flow monitoring for MPLS-over UDP traffic on PTX series routers, see "Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers" on page 465. Starting in Junos OS Release 18.2R1, you can configure up to four collectors under a family for inline active flow monitoring.

This procedure applies to PTX3000 and PTX5000 routers that have third-generation FPCs installed, PTX10008 and the PTX1000 routers. Starting in Junos OS Release 18.4R1, you can configure inline flow monitoring for IPv4 and IPv6 on PTX10002 routers.

Starting in Junos OS Release 19.4R1, the PTX10002-60C supports inline flow monitoring for MPLS, MPLS-IPv4, MPLS-IPv6, and MPLS-over-UDP traffic. Both IPFIX and version 9 templates are supported.

> **NOTE:** Starting in Junos OS Release 17.3R1, the PTX1000 supports version 9 flow templates. Prior to Junos OS Release 17.3R1, the PTX1000 does not support version 9 flow templates.

Inline flow monitoring is implemented on the Logical CPU (LCPU). All the functions like flow creation, flow update, and flow records export are done by the LCPU. The flow records are sent out in either the IPFIX format or the version 9 format.

The following limitations and restrictions apply to the inline active flow monitoring feature in Junos OS:

- Flow records and templates cannot be exported if the flow collector is reachable through any management interface.
- The flow collector should be reachable through the default routing table (inet.0 or inet6.0). If the flow collector is reachable through a non-default VPN routing and forwarding table (VRF), flow records and templates cannot be exported.
- For an ingress provider-edge node where an MPLS label push occurs over a plain IPv4 or IPv6 packet, we recommend that you configure only an MPLS egress filter with the sample action. If you also configure an inet or inet6 egress filter with a sample action, only the MPLS-IPv4 or MPLS-IPv6 flow is created, but the packets are counted twice.
- For an egress provider-edge node where an IPv4 or IPv6 explicit NULL label is received, we recommend that you configure only an inet or intet6 ingress filter with the sample action. If you also configure an MPLS ingress filter with a sample action, only the IPv4 or IPv6 flow is created, but the packets are counted twice.
- Egress sampling is not performed when an MPLS pop is executed on an MPLS packet and the egressing packet is also an MPLS packet.
- The top label IP address is reported correctly only when the label position 1 is in the first place, that is, [1 x y]. The x and y values can be any number between 2 and 8.
- For directly connected interfaces, the next hop is reported as 0 for ingress sampling at the egress provider-edge node and for egress sampling.
- True outgoing interface (OIF) reporting is not supported for egress sampling.
- Starting with Junos OS Release 18.2R1, you can configure up to four collectors under a family for inline active flow monitoring. In previous releases of Junos OS, you could configure only one collector under a family for inline active flow monitoring.

Configuring the Template to Specify Output Properties

Configure a template to specify the output properties for the flow records:

1. Configure the template name.

```
[edit services flow-monitoring]
user@host# set (version-ipfix | version9) template template-name
```

2. (Optional) Configure the interval after which an active flow is exported.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
```

```
user@host# set flow-active-timeout seconds
```

3. (Optional) Configure the interval of activity that marks a flow as inactive.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-inactive-timeout seconds
```

4. (Optional) Configure the frequency at which the flow generator sends updates about template definitions to the flow collector. Specify either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set template-refresh-rate packets packets seconds seconds
```

5. (Optional) Configure the refresh rate in either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set option-refresh-rate packets packets seconds seconds
```

6. Specify the type of record that the template is used for.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set (ipv4-template | ipv6-template | mpls-template)
```

7. If you are monitoring MPLS flows, identify the types of MPLS flows.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set tunnel-observation [ipv4 | ipv6 | mpls-over-udp]
```

The **tunnel-observation** values enable the creation of the following types of flows:

- **ipv4**—MPLS-IPv4 flows
- **ipv6**—MPLS-IPv6 flows
- **mpls-over-udp**—MPLS-over-UDP flows

You can configure multiple values for **tunnel-observation**. Flows are created for only the deepest match. For example, if you configure both **ipv4** and **mpls-over-udp** and the traffic type is MPLS-over-UDP, flows are created for MPLS-over-UDP. If you configure **ipv4** but *do not* configure **mpls-over-udp** and the traffic type is MPLS-over-UDP, flows are created for MPLS-IPv4.

If the MPLS traffic type does *not* match any of the **tunnel-observation** values, then plain MPLS flows are created.

If you do not configure **tunnel-observation**, plain MPLS flows are created.

8. Enable the learning of next-hop addresses so that the true outgoing interface (OIF) is reported.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set nexthop-learning
```

Configuring the Sampling Instance

Configure a sampling instance:

1. Configure the sampling instance name.

```
[edit forwarding-options sampling]
user@host# set instance instance-name
```

2. Configure the protocol family for the sampling instance.

```
[edit forwarding-options sampling instance instance-name]
user@host# set family (inet | inet6 | mpls)
```

3. Set the ratio of the number of packets to be sampled. For example, if you specify a rate of 10, every tenth packet (1 packet out of 10) is sampled.

```
[edit forwarding-options sampling instance instance-name input]
user@host# set rate number
```

BEST PRACTICE: We recommend that you use a value of 1000 or higher for MPLS flows.

4. Specify the source address for the traffic to be sampled.

```
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output]
user@host# set inline-jflow source-address address
```

5. Specify the flow export rate of monitored packets in kpps.

```
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output]
user@host# set inline-jflow flow-export-rate rate
```

6. Specify the output address and port for a flow server.

```
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output]
user@host# set flow-server hostname port port-number
```

7. Specify the template to use with the sampling instance.

```
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output flow-server
  hostname]
user@host# set (version9 | version-ipfix) template template-name
```

Assigning the Sampling Instance to an FPC

- Assign the sampling instance to the FPC on which you want to implement flow monitoring.

```
[edit chassis]
user@host# set fpc slot-number sampling-instance instance-name
```

Configuring a Firewall Filter

Configure a firewall filter to specify the family of traffic to accept and sample.

1. Configure the firewall filter name and specify the family of traffic.

```
[edit firewall]
user@host# edit family (inet | inet6 | mpls) filter filter-name
```

2. Configure a term to sample and accept traffic.

```
[edit firewall family (inet | inet6 | mpls) filter filter-name]
user@host# set term term-name then accept
user@host# set term term-name then sample
```

Assigning the Firewall Filter to the Monitored Interface

- Assign the input firewall filter to the interface you want to monitor.

```
[edit interfaces]
user@host# set interface-name unit logical-unit-number family (inet |inet6 | mpls) filter input filter-name
```

The following is an example of the sampling configuration for an instance that supports inline flow monitoring on **family inet** and on **family inet6**:

```
[edit forwarding-options]
sampling {
    instance {
        sample-ins1 {
            input {
                rate 1;
            }
            family inet {
                output {
                    flow-server 2.2.2.2 {
                        port 2055;
                        version-ipfix {
                            template {
                                ipv4;
                            }
                        }
                    }
                    inline-jflow {
                        source-address 10.11.12.13;
                    }
                }
            }
            family inet6 {
                output {
                    flow-server 2.2.2.2 {
                        port 2055;
                        version-ipfix {
                            template {
                                ipv6;
                            }
                        }
                    }
                    interface sp-0/1/0 {
```

```
                source-address 10.11.12.13;
            }
        }
    }
  }
}
```

The following example shows the output format configuration:

```
services {
   flow-monitoring {
      version-ipfix {
         template ipv4 {
            flow-active-timeout 60;
            flow-inactive-timeout 60;
            ipv4-template;
            template-refresh-rate {
               packets 1000;
               seconds 10;
            }
            option-refresh-rate {
               packets 1000;
               seconds 10;
            }
         }
      }
   }
}
```

Release History Table

Release	Description
19.4R1	Starting in Junos OS Release 19.4R1, the PTX10002-60C supports inline flow monitoring for MPLS, MPLS-IPv4, MPLS-IPv6, and MPLS-over-UDP traffic. Both IPFIX and version 9 templates are supported.
18.4R1	Starting in Junos OS Release 18.4R1, you can configure inline flow monitoring for IPv4 and IPv6 on PTX10002 routers.
18.2R1	Starting in Junos OS Release 18.2R1, you can also configure inline flow monitoring on the PTX Series for MPLS, MPLS-IPv4, and MPLS-IPv6 traffic.
18.2R1	Starting in Junos OS Release 18.2R1, you can configure up to four collectors under a family for inline active flow monitoring.
17.3R1	Starting in Junos OS Release 17.3R1, the PTX1000 supports version 9 flow templates.

Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers

IN THIS SECTION

You can enable flow monitoring that reports the inner payload of MPLS-over-UDP flows on PTX Series routers. For more information, see the following:

MPLS-over-UDP Flow Monitoring Overview

Starting with Junos OS Release 18.1R1 on PTX Series routers with an FPC3, PTX10K-LC1101, PTX10K-LC1102, or PTX1000 card, you can perform inline flow monitoring for MPLS-over-UDP flows to look past the tunnel header to sample and report on the inner payload at both the transit and egress nodes of the tunnel.

Starting with Junos OS Release 19.4R1, on the PTX10002-60C, you can perform inline flow monitoring for MPLS-over-UDP flows to look past the tunnel header to sample and report on the inner payload at both the transit and egress nodes of the tunnel.

Starting with Junos OS Release 19.4R1, the PTX10002-60C supports inline flow monitoring for MPLS, MPLS-IPv4, MPLS-IPv6, and MPLS-over-UDP traffic. Both IPFIX and version 9 templates are supported. For a description of the fields included in the templates, see "Understanding Inline Active Flow Monitoring" on page 391. Only ingress sampling is supported.

Benefits of Using MPLS-Over-UDP Flow Monitoring

- Gather and export detailed information on even the original IPv4 or IPv6 payload of the MPLS-over-UDP flow.

Flow Monitoring Scenarios for MPLS-over-UDP

Monitoring for MPLS-over-UDP tunnels includes the following scenarios:

- The MPLS-over-UDP flow is carried through a full IP network, using IPv4 endpoints on PTX Series routers (see Figure 40 on page 466). The inner payload may be IPv4 or IPv6. Figure 41 on page 466 shows the encapsulated packet. Flow monitoring reports the inner IP header and payload, in addition to the tunnel and MPLS fields.

  You can enable ingress monitoring for the MPLS-over-UDP tunnel at its transit and egress nodes. For example, in Figure 40 on page 466, you can enable ingress monitoring on routers R4, R5, R6, and R7.

Figure 40: MPLS-over-UDP in Full IP Network

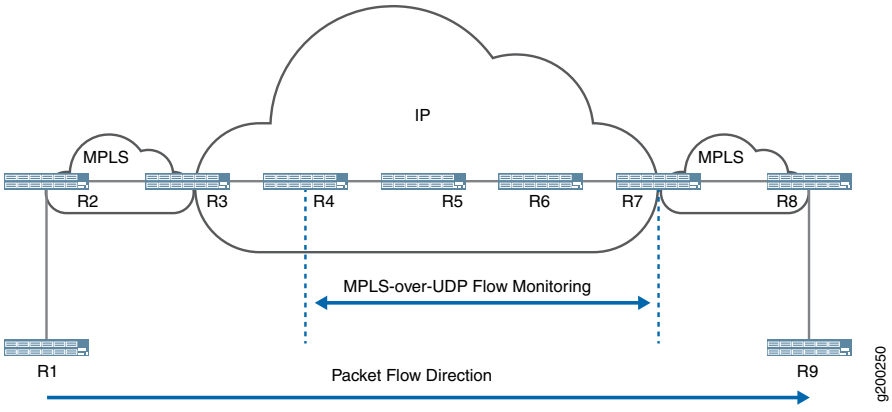
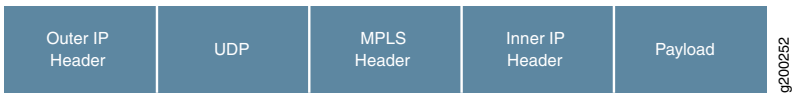


Figure 41: Encapsulated Packet for MPLS-over-UDP in Full IP Network



- The MPLS-over-UDP flow is carried through an IP-MPLS-IP network, using IPv4 endpoints on PTX Series routers (see Figure 42 on page 467). The inner payload may be IPv4 or IPv6. In the inner MPLS network, the MPLS-over-UDP flow is encapsulated in an RSVP-TE label-switched path (LSP). Figure 43 on page 467 shows the encapsulated packet. Flow monitoring reports the inner IP header and payload, in addition to the RSVP label, tunnel, and MPLS fields.

  You can enable ingress monitoring for the MPLS-over-UDP tunnel at its transit and egress nodes. For example, in Figure 42 on page 467, you can enable ingress monitoring on routers R4, R5, R6, R7, R8, and R9.

Figure 42: MPLS-over-UDP Over IP-MPLS-IP Network

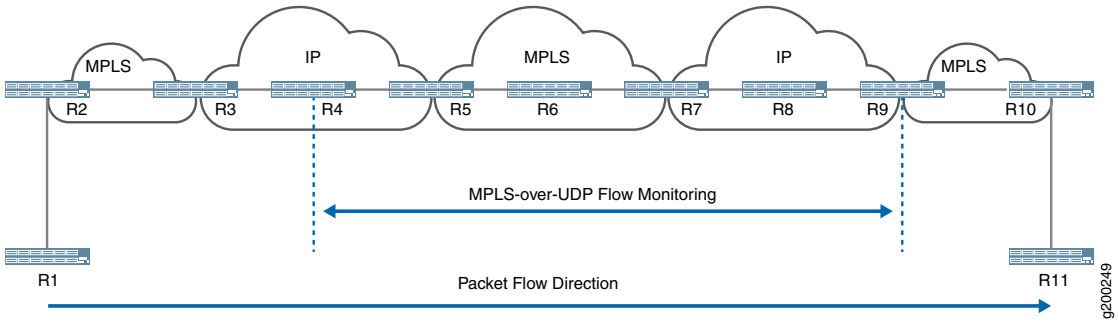


Figure 43: MPLS-over-UDP in RSVP-TE LSP Packet

Configuring Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers

IN THIS SECTION

Configuring inline active monitoring of MPLS-over-UDP flows includes the following tasks:

Configuring the Template to Specify Output Properties

Configure a template to specify the output properties for the flow records:

1. Configure the template name.

```
[edit services flow-monitoring]
user@host# set (version-ipfix | version9) template template-name
```

2. (Optional) Configure the interval after which an active flow is exported.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-active-timeout seconds
```

3. (Optional) Configure the interval of activity that marks a flow as inactive.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-inactive-timeout seconds
```

4. (Optional) Configure the frequency at which the flow generator sends updates about template definitions to the flow collector. Specify either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set template-refresh-rate packets packets seconds seconds
```

5. (Optional) Configure the refresh rate in either number of packets or number of seconds.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set option-refresh-rate packets packets seconds seconds
```

6. Enable flow monitoring of MPLS-over-UDP flows.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set tunnel-observation mpls-over-udp
```

7. Specify the template type.

   - If you are monitoring an MPLS-over-UDP flow that is carried through a full IP network (see Figure 40 on page 466), use the **ipv4-template**:

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set ipv4-template
```

- If you are monitoring an MPLS-over-UDP flow that is carried through an IP-MPLS-IP network (see Figure 42 on page 467):

  For the IP network transit and egress nodes (for example, R4, R5, R8, and R9 in Figure 42 on page 467), use the **ipv4-template** type.

  ```
  [edit services flow-monitoring (version-ipfix | version9) template template-name]
  user@host# set ipv4-template
  ```

  For the transit and egress nodes where the MPLS-over-UDP flow is encapsulated in an RSVP-TE LSP (for example R6 and R7 in Figure 42 on page 467), use one of the following templates:

  - Starting in Junos OS Release 18.2R1:

    ```
    [edit services flow-monitoring (version-ipfix | version9) template template-name]
    user@host# set mpls-template
    ```

  - In Junos OS Release 18.1:

    ```
    [edit services flow-monitoring (version-ipfix | version9) template template-name]
    user@host# set mpls-ipvx-template
    ```

8. Enable the learning of next-hop addresses so that the true outgoing interface (OIF) is reported.

   ```
   [edit services flow-monitoring (version-ipfix | version9) template template-name]
   user@host# set nexthop-learning
   ```

Configuring the Sampling Instance

Configure a sampling instance:

1. Configure the sampling instance name.

   ```
   [edit forwarding-options sampling]
   user@host# set instance instance-name
   ```

2. Configure the MPLS protocol family for the sampling instance.

   ```
   [edit forwarding-options sampling instance instance-name]
   ```

```
user@host# set family mpls
```

3. Set the ratio of the number of packets to be sampled. For example, if you specify a rate of 10, every tenth packet (1 packet out of 10) is sampled.

```
[edit forwarding-options sampling instance instance-name input]
user@host# set rate number
```

4. Specify the source address for the traffic to be sampled.

```
[edit forwarding-options sampling instance instance-name family mpls output]
user@host# set inline-jflow source-address address
```

5. Specify the flow export rate of monitored packets in kpps.

```
[edit forwarding-options sampling instance instance-name family mpls output]
user@host# set inline-jflow flow-export-rate rate
```

6. Specify the output address and port for a flow server.

```
[edit forwarding-options sampling instance instance-name family mpls output]
user@host# set flow-server hostname port port-number
```

7. Specify the template to use with the sampling instance.

```
[edit forwarding-options sampling instance instance-name family mpls output flow-server hostname]
user@host# set (version9 | version-ipfix) template template-name
```

Assigning the Sampling Instance to an FPC

- Assign the sampling instance to the FPC on which you want to implement flow monitoring.

```
[edit chassis]
user@host# set fpc slot-number sampling-instance instance-name
```

Configuring a Firewall Filter

Configure a firewall filter to accept and sample MPLS traffic.

1. Configure the MPLS firewall filter name.

```
[edit firewall]
user@host# edit family mpls filter filter-name
```

2. Configure a term to sample and accept MPLS packets.

```
[edit firewall family mpls filter filter-name]
user@host# set term term-name then accept
user@host# set term term-name then sample
```

Assigning the Firewall Filter to the Monitored Interface

- Assign the input firewall filter to the interface you want to monitor.

```
[edit interfaces]
user@host# set interface-name unit logical-unit-number family mpls filter input filter-name
```

Release History Table

Release	Description
19.4R1	Starting with Junos OS Release 19.4R1, on the PTX10002-60C, you can perform inline flow monitoring for MPLS-over-UDP flows to look past the tunnel header to sample and report on the inner payload at both the transit and egress nodes of the tunnel.
19.4R1	Starting with Junos OS Release 19.4R1, the PTX10002-60C supports inline flow monitoring for MPLS, MPLS-IPv4, MPLS-IPv6, and MPLS-over-UDP traffic. Both IPFIX and version 9 templates are supported.
18.1R1	Starting with Junos OS Release 18.1R1 on PTX Series routers with an FPC3, PTX10K-LC1101, PTX10K-LC1102, or PTX1000 card, you can perform inline flow monitoring for MPLS-over-UDP flows to look past the tunnel header to sample and report on the inner payload at both the transit and egress nodes of the tunnel.

Inline Active Flow Monitoring on IRB Interfaces

IN THIS SECTION

You can perform inline active flow monitoring for IPv4 and IPv6 traffic on the integrated routing and bridging (IRB) interfaces on PTX Series routers. For more information, see the following:

Inline Active Flow Monitoring on IRB Interfaces-Overview

Starting in Junos OS Release 19.1R1, on PTX Series routers, you can perform inline active flow monitoring for IPv4 and IPv6 traffic on the integrated routing and bridging (IRB) interfaces. Both IPFIX and version 9 templates for the flow monitoring are supported. For a description of the fields included in the templates, see “Understanding Inline Active Flow Monitoring” on page 391.

Understanding Inline Active Flow Monitoring on IRB interfaces

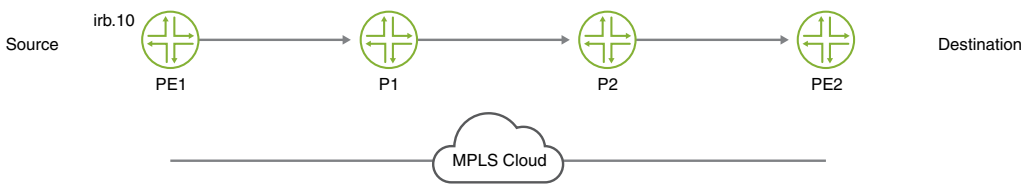
You can enable inline active flow monitoring by configuring the IPFIX or V9 templates on IRB interfaces.

You can configure inline active flow monitoring on IRB interfaces for the following scenarios:

Sampling on an IRB Interface with Traffic Routed to a Tunnelled Core

Figure 44 on page 472 illustrates sampling on an IRB interface where the traffic is routed to a tunnelled core, primarily an MPLS tunnel. The packets are entering irb.10 on which you can enable ingress sampling. The packets can be forwarded to a next hop which is not a part of any user-defined VLAN.

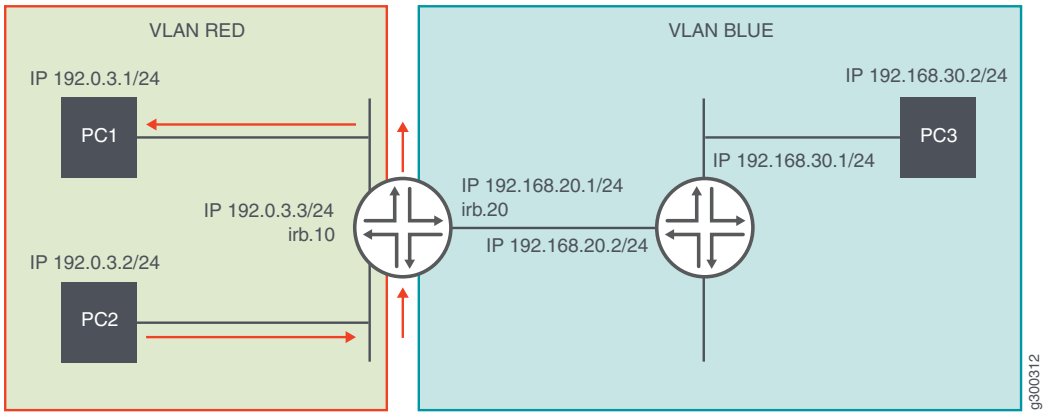
Figure 44: Sampling on an IRB Interface Routing Traffic to a Tunnelled Core



Layer 2 bridging and Layer 3 IP routing on an IRB interface

Figure 45 on page 473 illustrates the topology where Layer 2 bridging and Layer 3 IP routing are supported on the same interface.

Figure 45: Layer 2 Bridging and Layer 3 IP Routing on the Same IRB Interface

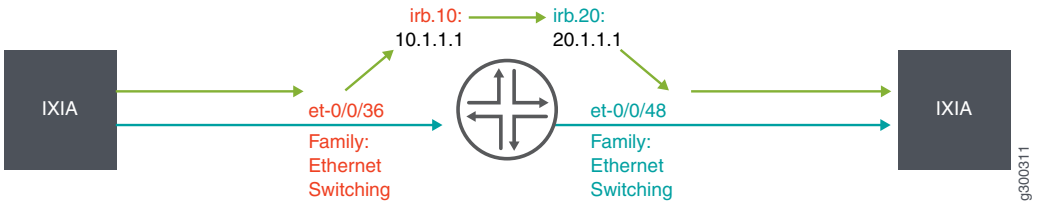


PC1 and PC2 are in VLAN RED (ID 10) and PC3 is in VLAN BLUE (ID 20).

For traffic moving from PC1 to PC3 or from PC2 to PC3, an IRB interface must be configured with a logical unit with an address in the subnet for VLAN RED and a logical unit with an address in the subnet for VLAN BLUE. The switch automatically directs routes to these subnets and uses these routes to forward traffic between VLANs. If traffic is flowing from VLAN RED to VLAN BLUE, you can configure ingress sampling on irb.10 and egress sampling on irb.20.

Figure 46 on page 473 illustrates sampling in a topology where Layer 2 bridging and Layer 3 IP routing are supported on the same interface. The interfaces, et-0/0/36.0 and irb.10 belong to VLAN ID 2. The interfaces, et-0/0/48 and irb.20 belong to VLAN ID 3. Packets are entering irb.10 and exiting on irb.20. Hence, you can configure ingress sampling on irb.10 and egress sampling on irb.20

Figure 46: Sampling on an IRB Interface Supporting Bridging and Routing



Configuring Inline Active Flow Monitoring on IRB Interfaces on PTX Series Routers

IN THIS SECTION

Configuring inline active monitoring on IRB interfaces includes the following tasks:

Configuring the Template to Specify Output Properties

Configure a template to specify the output properties for the flow records:

1. Configure the template name.

   ```
   [edit services flow-monitoring]
   user@host# set (version-ipfix | version9) template template-name
   ```

   For example:

   ```
   [edit services flow-monitoring]
   user@host# set version-ipfix template t1
   ```

2. (Optional) Configure the interval after which an active flow is exported.

   ```
   [edit services flow-monitoring (version-ipfix | version9) template template-name]
   user@host# set flow-active-timeout seconds
   ```

   For example:

   ```
   [edit services flow-monitoring (version-ipfix | version9) template template-name]
   user@host# set flow-active-timeout 10
   ```

3. (Optional) Configure the interval of activity that marks a flow as inactive.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-inactive-timeout seconds
```

For example:

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set flow-inactive-timeout 10
```

4. Specify the template type.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set template-name
```

For example:

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
user@host# set ipv4-template
```

Configuring the Sampling Instance

Configure a sampling instance:

1. Configure the sampling instance name.

```
[edit forwarding-options sampling]
user@host# set instance instance-name
```

For example:

```
[edit forwarding-options sampling]
user@host# set instance s1
```

2. Configure the protocol family for the sampling instance.

```
[edit forwarding-options sampling instance instance-name]
user@host# set family (inet | inet6 | mpls)
```

For example:

```
[edit forwarding-options sampling instance instance-name]
user@host# set family inet
```

3. Set the ratio of the number of packets to be sampled. For example, if you specify a rate of 10, every tenth packet (1 packet out of 10) is sampled.

```
[edit forwarding-options sampling instance instance-name input]
user@host# set rate number
```

For example:

```
[edit forwarding-options sampling instance instance-name input]
user@host# set rate 10
```

4. Specify the source address for the traffic to be sampled.

```
[edit forwarding-options sampling instance instance-name family inet output]
user@host# set inline-jflow source-address address
```

For example:

```
[edit forwarding-options sampling instance instance-name family inet output]
user@host# set inline-jflow source-address 10.10.0.1
```

5. Specify the output address and port for a flow server.

```
[edit forwarding-options sampling instance instance-name family inet output]
user@host# set flow-server hostname port port-number
```

For example:

```
[edit forwarding-options sampling instance instance-name family inet output]
user@host# set flow-server 10.10.10.2 port 2055
```

6. Specify the template to use with the sampling instance.

```
[edit forwarding-options sampling instance instance-name family inet output flow-server hostname]
user@host# set (version9 | version-ipfix) template template-name
```

For example:

```
[edit forwarding-options sampling instance instance-name family inet output]
user@host# set version-ipfix template t1
```

Assigning the Sampling Instance to an FPC

Assign the sampling instance to the FPC on which you want to implement flow monitoring.

```
[edit chassis]
user@host# set fpc slot-number sampling-instance instance-name
```

For example:

```
[edit chassis]
user@host# set fpc 0 sampling-instance s1
```

Configuring a Firewall Filter

Configure a firewall filter to specify the family of traffic to accept and sample

1. Configure the firewall filter name and specify the family of traffic.

   ```
   [edit firewall]
   user@host# edit family (inet | inet6 | mpls) filter filter-name
   ```

   For example:

   ```
   [edit firewall]
   user@host# edit family inet filter f2
   ```

2. Configure a term to sample and accept packets.

   ```
   [edit firewall family mpls filter filter-name]
   user@host# set term term-name then accept
   user@host# set term term-name then sample
   ```

   For example:

   ```
   [edit firewall family mpls filter filter-name]
   user@host# set term t1 then count c2
   user@host# set term t1 then accept
   user@host# set term t1 then sample
   ```

Associate a Layer 3 Interface with VLAN to Route Traffic

Assign the IRB Interface with the VLAN.

```
[edit vlans vlan-name]
user@host# set vlan-name vlan-id vlan-id-number
user@host# set vlan-name l3-interface l3-interface-name .logical-interface-number
```

For example:

```
[edit vlans vlan-name]
user@host# set vlan2 vlan-id 2
user@host# set vlan2 l3-interface irb.10
```

For example, if you are configuring inline flow monitoring using IRB while supporting layer 2 bridging and layer 3 IP routing on the same interface (See Figure 46 on page 473):

```
[edit vlans vlan-name]
user@host# set vlan-2 vlan-id 2
user@host# set vlan-2 l3-interface irb.10
user@host# set vlan-3 vlan-id 3
user@host# set vlan-3 l3-interface irb.20
```

Assigning the Firewall Filter to the Monitored Interface

Assign the input firewall filter to the interface you want to monitor. Also, configure the VLANs for which the interface can carry traffic.

```
[edit interfaces]
user@host# set interface-name unit logical-unit-number family (inet | inet6 | mpls) filter input filter-name address
```

For example, if you are configuring inline flow monitoring using IRB while supporting layer 2 bridging and layer 3 IP routing on the same interface (See Figure 46 on page 473):

```
[edit interfaces]
user@host# set et-0/0/36 unit 0 family ethernet-switching vlan members vlan2
user@host# set et-0/0/48 unit 0 family ethernet-switching vlan members vlan3
user@host# set et-0/0/60 unit 0 family inet address 10.10.10.1
user@host# set irb unit 1family inet filter input f2
user@host# set irb unit 1family inet address 10.1.1.1
user@host# set irb unit 2 family inet address 20.1.1.1
user@host# set irb unit 1 family inet address 10.1.1.1
user@host# set irb unit 2 family inet filter output f2
```

Release History Table

Release	Description
19.1R1	Starting in Junos OS Release 19.1R1, on PTX Series routers, you can perform inline active flow monitoring for IPv4 and IPv6 traffic on the integrated routing and bridging (IRB) interfaces.

Example: Configuring Inline Active Flow Monitoring on MX Series and T4000 Routers

IN THIS SECTION

Software and Hardware Requirements

- An MX Series router other than MX80
- Junos OS Release 13.2 or later.

NOTE:

- Junos OS Releases earlier than 13.2 also support inline active flow monitoring. However, some of the features discussed in this example are not supported on previous releases.
- You need Junos OS Release 14.2 or later for configuring inline active flow monitoring on T4000 routers with Type 5 FPC.

Overview

Inline active flow monitoring enables you to configure active sampling without making use of a services DPC. This topic explains the basic configuration for enabling inline active flow monitoring for IPv4 and IPv6 flows. You can also configure inline active flow monitoring for VPLS flows. To configure inline active

flow monitoring for VPLS flows, you must specify the **family** as **vpls** and include **vpls-template** at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.

Configuration

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.

Configuring Template Properties

```
set services flow-monitoring version9 template template1 flow-active-timeout 120
set services flow-monitoring version9 template template1 flow-inactive-timeout 60
set services flow-monitoring version9 template template1 template-refresh-rate packets 100
set services flow-monitoring version9 template template1 template-refresh-rate seconds 600
set services flow-monitoring version9 template template1 option-refresh-rate packets 100
set services flow-monitoring version9 template template1 option-refresh-rate seconds 600
set services flow-monitoring version9 template template1 ipv4-template
set services flow-monitoring version-ipfix template template-v61 flow-active-timeout 150
set services flow-monitoring version-ipfix template template-v61 flow-inactive-timeout 100
set services flow-monitoring version-ipfix template template-v61 template-refresh-rate seconds 30
set services flow-monitoring version-ipfix template template-v61 ipv6-template
```

Configuring a Sampling Instance

```
set forwarding-options sampling instance instance-1 input rate 1
set forwarding-options sampling instance instance-1 family inet output flow-server 10.50.1.2 port
    2055
set forwarding-options sampling instance instance-1 family inet output flow-server 10.50.1.2 version9
    template template1
set forwarding-options sampling instance instance-1 family inet output inline-jflow source-address
    10.50.1.100
set forwarding-options sampling instance instance-1 family inet output inline-jflow flow-export-rate
    10
set forwarding-options sampling instance instance-1 family inet6 output flow-server 10.50.1.2 port
    2055
set forwarding-options sampling instance instance-1 family inet6 output flow-server 10.50.1.2
    version-ipfix template template-v61
```

```
set forwarding-options sampling instance instance-1 family inet6 output inline-jflow source-address
  10.50.1.110
set forwarding-options sampling instance instance-1 family inet6 output inline-jflow flow-export-rate
  6
```

Configuring FPC Parameters

```
set chassis fpc 0 sampling-instance instance-1
set chassis fpc 0 inline-services flow-table-size ipv4-flow-table-size 8
set chassis fpc 0 inline-services flow-table-size ipv6-flow-table-size 7
```

Configuring FIrewall Filters

```
set firewall family inet filter inet-sample term t1 then sample
set firewall family inet filter inet-sample term t1 then accept
set firewall family inet6 filter inet6-sample term t1 then sample
set firewall family inet6 filter inet6-sample term t1 then accept
```

Configuring Interface Properties

```
set interfaces ge-0/0/4 unit 0 family inet filter input inet-sample
set interfaces ge-0/0/4 unit 0 family inet address 150.10.1.1/24
set interfaces ge-0/1/6 unit 0 family inet6 filter input inet6-sample
set interfaces ge-0/1/6 unit 0 family inet6 address 751b:b01:0:2::1/64
```

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

1. Configure the template properties for inline active flow monitoring.

```
[edit services flow-monitoring]
user@router1# set version9 template template1 ipv4-template
user@router1# set version9 template template1 flow-active-timeout 120
user@router1# set version9 template template1 flow-inactive-timeout 60
```

```
user@router1# set version9 template template1 template-refresh-rate packets 100
user@router1# set version9 template template1 option-refresh-rate packets 100
user@router1# set version-ipfix template template-v61 ipv6-template
user@router1# set version-ipfix template template-v61 flow-active-timeout 150
user@router1# set version-ipfix template template-v61 flow-inactive-timeout 100
user@router1# set version-ipfix template template-v61 template-refresh-rate seconds 30
user@router1# set version-ipfix template template-v61 option-refresh-rate seconds 30
```

2. Configure the sampling instance for inline active flow monitoring.

```
[edit forwarding-options sampling]
user@router1# set instance instance-1 input rate 1
user@router1# set instance instance-1 family inet output flow-server 10.50.1.2 port 2055
user@router1# set instance instance-1 family inet output flow-server 10.50.1.2 version9 template template1
user@router1# set instance instance-1 family inet output inline-jflow source-address 10.50.1.100
user@router1# set instance instance-1 family inet output inline-jflow flow-export-rate 10
user@router1# set instance instance-1 family inet6 output flow-server 10.50.1.2 port 2055
user@router1# set instance instance-1 family inet6 output flow-server 10.50.1.2 version-ipfix template
  template-v61
user@router1# set instance instance-1 family inet6 output inline-jflow source-address 10.50.1.110
user@router1# set instance instance-1 family inet6 output inline-jflow flow-export-rate 6
```

> **NOTE:** Until you complete the next step for associating the sampling instance with an FPC, the instance remains inactive and is marked **inactive** in the configuration.

3. Associate the sampling instance with the FPC on which you want to implement inline active flow monitoring, and also configure the hash table sizes.

NOTE:

In Junos OS releases earlier than Release 12.1, the following conditions are applicable for supporting backward compatibility when you configure the IPv4 and IPv6 flow table sizes for inline active flow monitoring:

- If you do not configure the **flow-table-size** statement at the **[edit chassis fpc *slot-number* inline-services]** hierarchy level, fifteen 256K entries are allocated by default for the IPv4 flow table and one 1K entry is allocated by default for the IPv6 flow table on the Packet Forwarding Engine.
- If you configure the **ipv4-flow-table-size *size*** statement at the **[edit chassis fpc *slot-number* inline-services flow-table-size]** hierarchy level and do not configure the **ipv6-flow-table-size *size*** statement at the **[edit chassis fpc *slot-number* inline-services flow-table-size]** hierarchy level, the number of units of 256K entries that you configure for the IPv4 flow table is allocated. For the IPv6 flow table, a default size of one 1K entry is allocated on the Packet Forwarding Engine.
- If you do not configure the **ipv4-flow-table-size *size*** statement at the **[edit chassis fpc slot-number inline-services flow-table-size]** hierarchy level and if you configure the **ipv6-flow-table-size *size*** statement at the **[edit chassis fpc slot-number inline-services flow-table-size]** hierarchy level, the number of units of 256K entries that you configure for the IPv6 flow table is allocated. For the IPv4 flow table, a default size of one 1K entry is allocated on the Packet Forwarding Engine.
- If you configure the sizes of both the IPv4 and IPv6 flow tables, the flow tables are created on the Packet Forwarding Engine based on the size that you specified.

NOTE: When you configure inline active flow monitoring for VPLS flows, include the **vpls-flow-table-size** statement.

```
[edit chassis]
user@router1# set fpc 0 sampling-instance instance-1
user@router1# set fpc 0 inline-services flow-table-size ipv4-flow-table-size 8
user@router1# set fpc 0 inline-services flow-table-size ipv6-flow-table-size 7
```

4. Configure firewall filters.

```
[edit firewall]
user@router1# set family inet filter inet-sample term t1 then sample
```

```
user@router1# set family inet filter inet-sample term t1 then accept
user@router1# set family inet6 filter inet6-sample term t1 then sample
user@router1# set family inet6 filter inet6-sample term t1 then accept
```

5. Associate the firewall filters configured in the previous step with the interfaces on which you want to set up inline active flow monitoring.

```
[edit interfaces]
set ge-0/0/4 unit 0 family inet filter input inet-sample
set ge-0/0/4 unit 0 family inet address 150.10.1.1/24
set ge-0/1/6 unit 0 family inet6 filter input inet6-sample
set ge-0/1/6 unit 0 family inet6 address 751b:b01:0:2::1/64
```

6. Commit the configuration.

```
[edit]
user@router1# commit
```

Results

From the configuration mode, confirm your configuration by entering **show services flow-monitoring**, **show forwarding-options sampling**, **show chassis fpc 0**, **show firewall**, and **show interfaces** commands. If the output does not display the intended configuration, repeat the instructions in the example to correct the configuration.

- **show services flow-monitoring**

```
version9 {
    template template1 {
        flow-active-timeout 120;
        flow-inactive-timeout 60;
        template-refresh-rate {
            packets 100;
            seconds 600;
        }
        option-refresh-rate {
            packets 100;
            seconds 600;
        }
        ipv4-template;
    }
}
```

```
    version-ipfix {
      template template-v61 {
        flow-active-timeout 150;
        flow-inactive-timeout 100;
        template-refresh-rate {
          seconds 30;
        }
        ipv6-template;
      }
    }
```

- **show forwarding-options sampling**

```
instance {
  instance-1 {
    input {
      rate 1;
    }
    family inet {
      output {
        flow-server 10.50.1.2 {
          port 2055;
          version9 {
            template {
              template1;
            }
          }
        }
        inline-jflow {
          source-address 10.50.1.100;
          flow-export-rate 10;
        }
      }
    }
    family inet6 {
      output {
        flow-server 10.50.1.2 {
          port 2055;
          version-ipfix {
            template {
              template-v61;
            }
          }
        }
```

```
            inline-jflow {
                source-address 10.50.1.110;
                flow-export-rate 6;
            }
        }
    }
  }
}
```

- **show chassis fpc 0**

```
sampling-instance instance-1;
    inline-services {
        flow-table-size {
            ipv4-flow-table-size 8;
            ipv6-flow-table-size 7;
        }
    }
```

- **show firewall**

```
family inet {
    filter inet-sample {
        term t1 {
            then {
                sample;
                accept;
            }
        }
    }
}
    family inet6 {
        filter inet6-sample {
            term t1 {
                then {
                    sample;
                    accept;
                }
            }
        }
    }
```

- **show interfaces**

```
...
ge-0/1/6 {
    vlan-tagging;
    unit 0 {
        family inet6 {
            filter {
                input inet6-sample;
            }
            address 751b:b01:0:2::1/64;
        }
    }
    }
    ge-0/0/4 {
        vlan-tagging;
        unit 0 {
            family inet {
                filter {
                    input inet-sample;
                }
                address 150.10.1.1/24;
            }
        }
    }
...
```

RELATED DOCUMENTATION

Understanding Inline Active Flow Monitoring | 391

Configuring Inline Active Flow Monitoring on MX80 and MX104 Routers | 455

CHAPTER 12

Sampling Data Using Flow Aggregation

IN THIS CHAPTER

Understanding Flow Aggregation

You can collect an aggregate of sampled flows and send the aggregate to a specified host that runs either the cflowd application available from CAIDA (http://www.caida.org) or the newer version 9 format defined in RFC 3954, *Cisco Systems NetFlow Services Export Version 9*. Before you can perform flow aggregation, the routing protocol process must export the autonomous system (AS) path and routing information to the sampling process.

By using flow aggregation, you can obtain various types of byte and packet counts of flows through a router. The application collects the sampled flows over a period of 1 minute. At the end of the minute, the

number of samples to be exported are divided over the period of another minute and are exported over the course of the same minute.

You configure flow aggregation in different ways, depending on whether you want to export flow records in cflowd version 5 or 8 format, or the separate version 9 format. The latter allows you to sample MPLS, IPv4, IPv6, and peer AS billing traffic. You can also combine configuration statements between the MPLS and IPv4 formats.

> **NOTE:** When PIC-based sampling is enabled, collection of flow statistics for sampled packets on flows in virtual private networks (VPNs) is also supported. No additional CLI configuration is required.

RELATED DOCUMENTATION

- Enabling Flow Aggregation | **489**
- Configuring Flow Aggregation on MX, M and T Series Routers and NFX250 to Use Version 5 or Version 8 cflowd | **490**
- Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**
- Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | **540**
- Logging cflowd Flows on M and T Series Routers Before Export | **542**

Enabling Flow Aggregation

Before you can perform flow aggregation, the routing protocol process must export the autonomous system (AS) path and routing information to the sampling process. To enable the export of AS path and the routing information to the sampling process, one or more of the following needs to be configured:

- At the **[edit forwarding-options]** hierarchy level (for routing instances, at the **[edit routing-instance *routing-instance-name* forwarding-options]** hierarchy level), configure **sampling family** or **sampling output** or **sampling instance** or **monitoring** or **accounting**.
- At the **[edit routing-options]** hierarchy level (for routing instances, at the **[edit routing-instance *routing-instance-name* routing-options]** hierarchy level), configure **route record**.
- At the **[edit chassis fpc *slot-number* pic *pic-number* adaptive-services service-package extension-provider]** hierarchy level, configure **forwarding-db-size**.

RELATED DOCUMENTATION

- Understanding Flow Aggregation | 488
- Configuring Flow Aggregation on MX, M and T Series Routers and NFX250 to Use Version 5 or Version 8 cflowd | 490
- Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495
- Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | 540
- Configuring Traffic Sampling on MX, M and T Series Routers | 375
- Logging cflowd Flows on M and T Series Routers Before Export | 542

Configuring Flow Aggregation on MX, M and T Series Routers and NFX250 to Use Version 5 or Version 8 cflowd

To enable the collection of cflowd version 5 or version 8 flow formats, include the **flow-server** statement:

```
flow-server hostname {
    aggregation {
        autonomous-system;
        destination-prefix;
        protocol-port;
        source-destination-prefix {
            caida-compliant;
        }
        source-prefix;
    }
    autonomous-system-type (origin | peer);
    (local-dump | no-local-dump);
    port port-number;
    version format;
}
```

You can include this statement at the following hierarchy levels:

- **[edit forwarding-options sampling family (inet | inet6 | mpls) output]**
- **[edit forwarding-options sampling instance *instance-name* output]**
- **[edit forwarding-options accounting *name* output cflowd *hostname*]**

You must configure the **family inet** statement on logical interface **unit 0** on the monitoring interface, as in the following example:

```
[edit interfaces]
sp-3/0/0 {
    unit 0 {
        family inet {
            ...
        }
    }
}
```

NOTE: Boot images for monitoring services interfaces are specified at the **[edit chassis images pic]** hierarchy level. You must enable the NTP client to make the cflowd feature operable, by including the following configuration:

```
[edit system]
ntp {
    boot-server ntp.example.com;
    server 172.17.28.5;
}
processes {
    ntp enable;
}
```

For more information, see the *Junos OS Administration Library*.

You can also configure cflowd version 5 for flow-monitoring applications by including the **cflowd** statement at the **[edit forwarding-options monitoring *name* family inet output]** hierarchy level:

```
cflowd hostname {
    port port-number;
}
```

The following restrictions apply to cflowd flow formats:

- You can configure up to one version 5 and one version 8 flow format at the **[edit forwarding-options accounting *name* output]** hierarchy level.
- You can configure up to eight version 5 or one version 8 flow format at the **[edit forwarding-options sampling family (inet | inet6 | mpls) output]** hierarchy level for Routing Engine-based sampling by including the **flow-server** statement. In contrast, PIC-based sampling allows you to specify one cflowd version 5 server and one version 8 server simultaneously. However, the two cflowd servers must have different IP addresses.

- You can configure up to eight version 5 flow formats at the **[edit forwarding-options monitoring *name* output]** hierarchy level. Version 8 flow formats and aggregation are not supported for flow-monitoring applications.
- Outbound Routing Engine traffic is not sampled. A firewall filter is applied as output on the egress interface, which samples packets and exports the data. For transit traffic, egress sampling works correctly. For internal traffic, the next hop is installed in the Packet Forwarding Engine but sampled packets are not exported.
- Flows are created on the monitoring PIC only after the route record resynchronization operation is complete, which is 60 seconds after the PIC comes up. Any packets sent to the PIC are dropped until the synchronization process is complete.
- The configuration includes a proprietary v5 extension template for supporting 4-byte AS information in flow records. Its template version is set to 500, indicating it to be proprietary. All other fields remain the same; the source AS and destination AS are each 4 bytes long, rather than 2 bytes as in the traditional v5 template. This option is available at the **[edit forwarding-options sampling family inet output flow-server *server-name* version]** hierarchy level.

In the **cflowd** statement, specify the name or identifier of the host that collects the flow aggregates. You must also include the User Datagram Protocol (UDP) port number on the host and the version, which gives the format of the exported cflowd aggregates. To collect cflowd records in a log file before exporting, include the **local-dump** statement.

> **NOTE:** You can specify both host (cflowd) sampling and port mirroring in the same configuration; however, only one action takes effect at any one time. Port mirroring takes precedence. For more information, see "Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers" on page 547.

For cflowd version 8 only, you can specify aggregation of specific types of traffic by including the **aggregation** statement. This conserves memory and bandwidth by enabling cflowd to export targeted flows rather than all aggregated traffic. To specify a flow type, include the **aggregation** statement:

```
aggregation {
    autonomous-system;
    destination-prefix;
    protocol-port;
    source-destination-prefix {
        caida-compliant;
    }
    source-prefix;
}
```

You can include this statement at the following hierarchy levels:

- **[edit forwarding-options sampling family (inet | inet6 | mpls) output flow-server *hostname*]**
- **[edit forwarding-options accounting *name* output cflowd *hostname*]**

The **autonomous-system** statement configures aggregation by the AS number; this statement might require setting the separate cflowd **autonomous-system-type** statement to include either **origin** or **peer** AS numbers. The **origin** option specifies to use the origin AS of the packet source address in the Source Autonomous System cflowd field. The **peer** option specifies to use the peer AS through which the packet passed in the Source Autonomous System cflowd field. By default, cflowd exports the origin AS number.

The **destination-prefix** statement configures aggregation by the destination prefix only.

The **protocol-port** statement configures aggregation by the protocol and port number; requires setting the separate **cflowd port** statement.

The **source-destination-prefix** statement configures aggregation by the source and destination prefix. Version 2.1b1 of CAIDA's cflowd application does not record source and destination mask length values in compliance with CAIDA's *cflowd Configuration Guide*, dated August 30, 1999. If you configure the **caida-compliant** statement, the Junos OS complies with Version 2.1b1 of cflowd. If you do not include the **caida-compliant** statement in the configuration, the Junos OS records source and destination mask length values in compliance with the *cflowd Configuration Guide*.

The **source-prefix** statement configures aggregation by the source prefix only.

Collection of sampled packets in a local ASCII file is not affected by the **cflowd** statement.

The following commands enable Routing Engine- and PIC-based sampling at the **set forwarding options sampling** hierarchy level:

- **set input rate *rate***
- **set input run-length *length***
- **set family inet output flow-server *flowcollector* port *udp port***
- **set family inet output flow-server *flowcollector* no-local-dump**
- **set family inet output flow-server *flowcollector* version <5/8>**

The following commands enable Routing Engine- and PIC-based sampling at the **set interfaces** hierarchy level:

- *interface to be sampled* unit *unit* family inet filter *input/output filtername*

The following commands enable Routing Engine- and PIC-based sampling at the **set firewall family** hierarchy level:

- **set inet filter *filtername* term 1 then count *filtername*ing**
- **set inet filter *filtername* term 1 then sample**
- **set inet filter *filtername* term 1 then accept**

The following command enables PIC-based sampling at the **set forwarding options sampling** hierarchy level:

- **set family inet output interface *sp-*/*/** source address *source address***

The following example shows a PIC-based flow aggregation configuration using version 5:

```
family inet {
    output {
        flow-inactive-timeout 15;
        flow-active-timeout 60;
        flow-server 203.0.113.165 {
            port 9996;
            version 5;
        }
        interface sp-2/2/0 {
        engine-id 4;
        source-address 203.0.113.126;
    }
}
```

The following example shows an Routing Engine-based flow aggregation configuration using version 5:

```
family inet {
    output {
        flow-inactive-timeout 15;
        flow-active-timeout 60;
        flow-server 203.0.113.165 {
            port 9996;
            source-address 203.0.113.126;
            version 5;
        }
    }
```

RELATED DOCUMENTATION

Understanding Flow Aggregation | **488**

Enabling Flow Aggregation | **489**

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates

IN THIS SECTION

Use of version 9 flow template enables you to define a flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic. Templates and the fields included in the template are transmitted to the collector periodically, and the collector does not affect the router configuration.

NOTE: Version 9 requires that you install a services PIC, such as the Adaptive Services PIC or MS-PIC in the router. On MX Series routers, the MS-DPC fulfills this requirement. For more information on determining which services PIC is suitable for your router, see *Enabling Service Packages* or the appropriate hardware documentation.

NOTE: If multiple protocol families are configured for a particular flow collector, the export packets originates from multiple Source IDs, with each Source ID corresponding to a particular protocol. The multiple Source IDs do not indicate that the export packets are originating from multiple Service PICs.

The following sections contain additional information:

Configuring the Traffic to Be Sampled

To specify sampling of IPv4, IPv6, MPLS, or peer AS billing traffic, include the appropriate configuration of the **family** statement at the **[edit forwarding-options sampling]** hierarchy level:

```
[edit forwarding-options]
sampling {
family (inet | inet6 | mpls);
}
```

You can include **family inet** ,**family inet6**, or **family mpls**.

> **NOTE:** If you specify sampling for peer AS billing traffic, the **family** statement supports only IPv4 and IPv6 traffic (inet or inet6). Peer AS billing traffic is enabled only at the global instance hierarchy level and is not available for per Packet Forwarding Engine instances.

After you specify the family of traffic to be sampled, configure the sampling parameters such as:

- Maximum packet length (beyond which the packets are truncated).
- Maximum packets to be sampled per second (beyond which the packets are dropped).
- Rate (for example, if you specify 10, every 10th packet is sampled).
- Run length (which specifies the number of packets to be sampled after the trigger; that is, if the **rate** is set to 10 and **run-length** to 5, five packets starting at the 10th packet are sampled).

```
[edit forwarding-options sampling]
input {
maximum-packet-length bytes
max-packets-per-second number;
rate number;
run-length number;
}
```

Configuring the Version 9 Template Properties

To define the Version 9 templates, include the following statements at the **[edit services flow-monitoring version9]** hierarchy level:

```
[edit services flow-monitoring version9]
template template-name {
```

```
    options-template-id
    template-id
    source-id
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    option-refresh-rate packets packets seconds seconds;
    template-refresh-rate packets packets seconds seconds;
    (ipv4-template | ipv6-template | mpls-ipv4-template | mpls-template | peer-as-billing-template) {
      label-position [ positions ];
    }
  }
```

The following details apply to the configuration statements:

- You assign each template a unique name by including the **template *template-name*** statement.
- You then specify each template for the appropriate type of traffic by including the **ipv4-template**, **ipv6-template**, **mpls-ipv4-template**, or **mpls-template**.
- If the template is used for MPLS traffic, you can also specify up to three label positions for the MPLS header label data by including the **label-position** statement; the default values are **[1 2 3]**.
- Within the template definition, you can optionally include values for the **flow-active-timeout** and **flow-inactive-timeout** statements. These statements have specific default and range values when they are used in template definitions; the default is 60 seconds and the range is from 10 through 600 seconds. Values you specify in template definitions override the global timeout values configured at the **[edit forwarding-options sampling family (inet | inet6 | mpls) output flow-server]** hierarchy level.
- You can also include settings for the **option-refresh-rate** and **template-refresh-rate** statements within a template definition. For both of these properties, you can include a timer value (in seconds) or a packet count (in number of packets). For the **seconds** option, the default value is 60 and the range is from 10 through 600. For the **packets** option, the default value is 4800 and the range is from 1 through 480,000.
- To filter IPv6 traffic on a media interface, the following configuration is supported:

```
interfaces interface-name {
  unit 0 {
    family inet6 {
      sampling {
        input;
        output;
      }
    }
  }
}
```

Customizing Template ID, Observation Domain ID, and Source ID for Version 9 Flow Templates

Starting in Junos OS Release 14.1, you can define a Version 9 flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic. Templates and the fields included in the template are transmitted to the collector periodically,and the collector does not affect the router configuration. You can specify the unique identifier for the version 9 and IPFIX templates. The identifier of a template is locally unique within a combination of a transport session and an observation domain. Template IDs 0 through 255 are reserved for template sets, options template sets, and other sets for future use. Template IDs of data sets are numbered from 256 through 65535. Typically, this information element or field in the template is used to define the characteristics or properties of other information elements in a template. After a restart of the export process of templates is performed, you can reassign template IDs.

This functionality to configure template ID, options template ID, observation domain ID, and source ID is supported on all routers with MPCs.

> **NOTE:** The template IDs that include MPLS and MPLS-IPv4 template ID are applicable for IPFIX only. The V9 format carries a different template ID.

The corresponding data sets and option data sets contain the value of the template IDs and options template IDs respectively in the set ID field. This method enables the collector to match a data record with a template record.

For more information about specifying the source ID, observation domain ID, template ID, and options template ID for version 9 and IPFIX flows, see "Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows" on page 527 and "Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows" on page 531.

Restrictions

The following restrictions apply to version 9 templates:

- You cannot apply the two different types of flow aggregation configuration at the same time.
- Flow export based on an **mpls-ipv4** template assumes that the IPv4 header follows the MPLS header. In the case of Layer 2 VPNs, the packet on the provider router (P router) looks like this:

  ```
  MPLS | Layer 2 Header | IPv4
  ```

  In this case, **mpls-ipv4** flows are not created on the PIC, because the IPv4 header does not directly follow the MPLS header. Packets are dropped on the PIC and are accounted as parser errors.

- Outbound Routing Engine traffic is not sampled. A firewall filter is applied as output on the egress interface, which samples packets and exports the data. For transit traffic, egress sampling works correctly. For internal traffic, the next hop is installed in the Packet Forwarding Engine but sampled packets are not exported.
- Flows are created on the monitoring PIC only after the route record resynchronization operation is complete, which is 60 seconds after the PIC comes up. Any packets sent to the PIC are dropped until the synchronization process is complete.

> **NOTE:** Because the forwarding of a packet that arrives with MPLS labels is performed based on the MPLS label and not based on the IP address contained in the packet, the packet is sampled at the output interface with the MPLS label that was popped not being available at the time of sampling. In such a case, depending on the incoming interface (IIF), the VRF index is identified and the route for the sampled packet is determined in the VRF table. Because a specific route is not available in the VRF that is different from the VRF on which the packet is received, the Output Interface Index, Source Mask, and Destination Mask fields are incorrectly populated. This behavior occurs when an IPv4 template is applied as a firewall filter on an egress interface with sample as the action.

Fields Included in Each Template Type

The following fields are common to all template types:

- Input interface
- Output interface
- Number of bytes
- Number of packets
- Flow start time
- Flow end time

The IPv4 template includes the following specific fields:

- IPv4 Source Address
- IPv4 Destination Address
- L4 Source Port
- L4 Destination Port
- IPv4 ToS
- IPv4 Protocol

- ICMP type and code
- TCP Flags
- IPv4 Next Hop Address
- Source autonomous system (AS) number
- Destination AS number

The IPv6 template includes the following specific fields:

- IPv6 Source Address and Mask
- IPv6 Destination Address and Mask
- L4 Source Port
- L4 Destination Port
- IPv6 ToS
- IPv6 Protocol
- TCP Flags
- IP Protocol Version
- IPv6 Next Hop Address
- Egress Interface Information
- Source Autonomous System (AS) number
- Destination AS number

The MPLS template includes the following specific fields:

- MPLS Label #1
- MPLS Label #2
- MPLS Label #3
- MPLS EXP Information
- FEC IP Address

The MPLS-IPv4 template includes all the fields found in the IPv4 and MPLS templates.

The peer AS billing template includes the following specific fields:

- IPv4 Class of Service (ToS)
- Ingress Interface
- BGP IPv4 Next Hop Address
- BGP Peer Destination AS Number

MPLS Sampling Behavior

This section describes the behavior when MPLS sampling is used on egress interfaces in various scenarios (label pop or swap) on provider routers (P routers). For more information on configuration and background specific to MPLS applications, see the *MPLS Applications User Guide*.

- You configure MPLS sampling on an egress interface on the P router and configure an MPLS flow aggregation template. The route action is label *pop* because penultimate hop popping (PHP) is enabled.

  With the current capability of applying MPLS templates, MPLS flows are created.

- As in the first case, you configure MPLS sampling on an egress interface on the P router and configure an MPLS flow aggregation template. The route action is label swap and the swapped label is 0 (explicit null).

  The resulting behavior is that MPLS packets are sent to the PIC. The flow being sampled corresponds to the label before the swap.

- You configure a Layer 3 VPN network, in which a customer edge router (CE-1) sends traffic to a provider edge router (PE-A), through the P router, to a similar provider edge router (PE-B) and customer edge router (CE-2) on the remote end.

  The resulting behavior is that you cannot sample MPLS packets on the PE-A to P router link.

Verification

To verify the configuration properties, you can use the **show services accounting aggregation template template-name *name*** operational mode command.

All other **show services accounting** commands also support version 9 templates, except for **show services accounting flow-detail** and **show services accounting aggregation aggregation-type**. For more information about operational mode commands, see the CLI Explorer.

Examples: Configuring Version 9 Flow Templates

The following example shows a version 9 template configuration:

```
services {
    flow-monitoring {
        version9 {
            template ip-template {
                flow-active-timeout 20;
                flow-inactive-timeout 120;
                ipv4-template;
            }
            template mpls-template-1 {
```

```
            mpls-template {
                label-position [1 3 4];
            }
        }
        template mpls-ipv4-template-1 {
            mpls-ipv4-template {
                label-position [1 5 7];
            }
        }
        template vpls-template-1 {
            vpls-template;
            }
        }
    }
  }
}
```

The following example shows a firewall filter configuration for MPLS traffic:

```
firewall {
    family mpls {
        filter mpls_sample {
            term default {
                then {
                    accept;
                    sample;
                }
            }
        }
    }
}
```

The following example applies the MPLS sampling filter on a networking interface and configures the AS PIC to accept both IPv4 and MPLS traffic:

```
interfaces {
    at-0/1/1 {
        unit 0 {
            family mpls {
                filter {
                    input mpls_sample;
                }
            }
```

```
        }
    }
    sp-7/0/0 {
        unit 0 {
            family inet;
            family mpls;
        }
    }
}
```

The following example applies the MPLS version 9 template to the sampling output and sends it to the AS PIC:

```
forwarding-options {
    sampling {
        input {
            family mpls {
                rate 1;
            }
        }
        family mpls {
            output {
            flow-active-timeout 60;
            flow-inactive-timeout 30;
            flow-server 192.0.2.4 {
                port 2055;
                version9 {
                    template mpls-ipv4-template-1;
                }
            }
            interface sp-7/0/0 {
                source-address 198.51.100.1;
            }
            }
        }
    }
}
```

The following example shows a firewall filter configuration for the peer AS billing traffic:

```
firewall {
    family inet {
        filter peer-as-filter {
```

```
        term 0 {
            from {
                destination-class dcu-1;
                interface ge-2/1/0;
                forwarding-class class-1;
            }
            then count count_team_0;
        }
        }
        term 1 {
            from {
                destination-class dcu-2;
                interface ge-2/1/0;
                forwarding-class class-1;
            }
            then count count_team_1;
        }
        term 2 {
            from {
                destination-class dcu-3;
                interface ge-2/1/0;
                forwarding-class class-1;
            }
            then count count_team_2;
        }
      }
    }
}
```

The following example applies the peer AS firewall filter as a filter attribute under the forwarding-options hierarchy for CoS-level data traffic usage information collection:

```
forwarding-options {
   family inet {
      filter output peer-as-filter;
   }
}
```

The following example applies the peer AS DCU policy options to collect usage statistics for the traffic stream for as-path ingressing at a specific input interface with the firewall configuration hierarchy applied as Forwarding Table Filters (FTFs). The configuration functionality with CoS capability can be achieved through FTFs for destination-class usage with forwarding-class for specific input interfaces:

```
policy-options {
   policy-statement P1 {
         from {
            protocol bgp;
            neighbor 10.2.25.5; #BGP router configuration;
            as-path AS-1; #AS path configuration;
         }
         then destination-class dcu-1; #Destination class configuration;
   }
   policy-statement P2 {
         from {
            neighbor 203.0.113.5;
            as-path AS-2;
         }
         then destination-class dcu2;
   }
   policy-statement P3 {
         from {
            protocol bgp;
            neighbor 192.0.2.129;
            as-path AS-3;
         }
         then destination-class dcu3;
   }
   as-path AS-1 3131:1111:1123;
   as-path AS-2 100000;
   as-path AS-3 192:29283:2;
}
```

The following example applies the vpls version 9 template to enable sampling of traffic for billing purposes:

```
forwarding-options {
   sampling {
      }
      input {
         rate 1;
      }
      family inet {
         output {
            flow-server 10.209.15.58 {
               port 300;
               version9 {
                  template {
                     peer-as;
```

```
                }
              }
            }
            interface sp-5/2/0 {
              source-address 203.0.113.133;
            }
          }
        }
      }
    }
    family inet {
      filter {
        output peer-as-filter;
      }
    }
```

Release History Table

Release	Description
14.1	Starting in Junos OS Release 14.1, you can define a Version 9 flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic.

RELATED DOCUMENTATION

- Understanding Flow Aggregation | **488**
- Enabling Flow Aggregation | **489**
- Configuring Flow Aggregation on MX, M and T Series Routers and NFX250 to Use Version 5 or Version 8 cflowd | **490**
- Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**
- Configuring Traffic Sampling on MX, M and T Series Routers | **375**

Configuring Flow Aggregation on PTX Series Routers to Use Version 9 Flow Templates

IN THIS SECTION

You can define a flow record template suitable for IPv4 traffic or IPv6 traffic using a version 9 flow template. Templates and the fields included in the template are transmitted to the collector periodically. The collector does not affect the router configuration. You can define template refresh rate, flow active timeout and inactive timeout.

If flow records are being sent for multiple protocol families (for example, for IPv4 and IPv6), each protocol family flow will have a unique Observation Domain ID.

The following sections contain additional information:

Configuring the Version 9 Template Properties

To define the version 9 templates, include the following statements at the **[edit services flow-monitoring version9]** hierarchy level:

```
[edit services flow-monitoring version9]
template name {
   options-template-id
   template-id
   observation-domain-id
   flow-active-timeout seconds;
   flow-inactive-timeout seconds;
   option-refresh-rate packets packets seconds seconds;
   template-refresh-rate packets packets seconds seconds;
   (ipv4-template | ipv6-template);
```

```
}
```

The following details apply to the configuration statements:

- You assign each template a unique name by including the **template *name*** statement.
- You specify each template for the appropriate type of traffic by including the **ipv4-template** or **ipv6–template**.
- Within the template definition, you can optionally include values for the **flow-active-timeout** and **flow-inactive-timeout** statements. These statements have specific default and range values when they are used in template definitions; the default is 60 seconds and the range is from 10 through 600 seconds.
- You can also include settings for the **option-refresh-rate** and **template-refresh-rate** statements within a template definition. For both of these properties, you can include a timer value (in seconds) or a packet count (in number of packets). For the **seconds** option, the default value is 600 and the range is from 10 through 600. For the **packets** option, the default value is 4800 and the range is from 1 through 480,000.
- To filter IPv6 traffic on a media interface, the following configuration is supported:

```
interfaces interface-name {
    unit 0 {
        family inet6 {
            sampling {
                input;
                output;
            }
        }
    }
}
```

Restrictions

The following restrictions apply to version 9 templates:

- Outbound Routing Engine traffic is not sampled. A firewall filter is applied as output on the egress interface, which samples packets and exports the data. For transit traffic, egress sampling works correctly. For internal traffic, the next hop is installed in the Packet Forwarding Engine but sampled packets are not exported.
- Flows are created only after the route record resynchronization operation is complete, which takes 120 seconds.

Customizing Template ID, Observation Domain ID, and Source ID for Version 9 flow Templates

> **NOTE:** For PTX Series routers with third generation FPCs installed, the FPC's slot number is used for the observation domain ID.

Use of version 9 flow templates allow you to define a flow record template suitable for IPv4 traffic or IPv6 traffic. Templates and the fields included in the template are transmitted to the collector periodically, and the collector does not need to be aware of the router configuration. Template IDs 0 through 255 are reserved for template sets, options template sets, and other sets for future use. Template IDs of data sets are numbered from 256 through 65535. Typically, this information element or field in the template is used to define the characteristics or properties of other information elements in a template. After a restart of the export process of templates is performed, template IDs can be reassigned.

The corresponding data sets and option data sets contain the value of the template IDs and options template IDs respectively in the set ID field. This method enables the collector to match a data record with a template record.

Fields Included in the IPv4 Templates for PTX Series Routers

Table 88 on page 509 shows the fields that are available in the IPv4 templates.

Table 88: IPv4 Template Fields

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12
IPv4 TOS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
Source AS	16

Table 88: IPv4 Template Fields *(continued)*

Field	Element ID
Destination AS	17
BGP Next Hop Address	18
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv4 Next Hop Address	15
IPv4 Source Mask	9
IPv4 Destination Mask	13
TCP Flags	6
IP Protocol Version	60

Fields Included in the IPv6 Templates for PTX Series Routers

Table 89 on page 510 shows the fields that are available in the IPv6 templates.

Table 89: IPv6 Template Fields

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 TOS	5
IPv6 Protocol	4
L4 Source Port	7

Table 89: IPv6 Template Fields *(continued)*

Field	Element ID
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
Source AS	16
Destination AS	17
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv6 Next Hop Address	62
IPv6 Source Mask	29
IPv6 Destination Mask	30
TCP Flags	6
IP Protocol Version	60

Verification

The following show commands are supported for version 9:

- **show services accounting flow inline-jflow fpc-slot *fpc-slot***
- **show services accounting errors inline-jflow fpc-slot *fpc-slot***
- **show services accounting status inline-jflow fpc-slot *fpc-slot***

Example: Configuring an version 9 Flow Templates and Flow Sampling

The following is a sample version 9 template configuration:

```
services {
    flow-monitoring {
        version9 {
            template ipv4 {
                flow-active-timeout 60;
                flow-inactive-timeout 70;
                template-refresh-rate seconds 30;
                option-refresh-rate seconds 30;
                ipv4-template;
            }
        }
    }
}
```

```
chassis;
    fpc 0 {
        sampling-instance s1;
    }
```

The following example applies the version 9 template to enable sampling of traffic for billing:

```
forwarding-options {
    sampling {
        instance {
            s1 {
                input {
                    rate 10;
                }
                family inet {
                    output {
                        flow-server 11.11.4.2 {
                            port 2055;
                            version9 {
                                template {
                                    ipv4;
                                }
                            }
                        }
                        inline-jflow {
```

```
                source-address 11.11.2.1;
            }
        }
    }
  }
 }
}
```

RELATED DOCUMENTATION

- Configuring Inline Active Flow Monitoring on PTX Series Routers | **458**
- version9 (Flow Monitoring) | **1216**
- ipv4-template | **1013**
- ipv6-template | **1017**

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices

IN THIS SECTION

Use of IPFIX allows you to define a flow record template suitable for IPv4 traffic or IPv6 traffic. Templates are transmitted to the collector periodically, and the collector does not affect the router configuration. You can define template refresh rate, flow active timeout and inactive timeout.

If flow records are being sent for multiple protocol families (for example, for IPv4 and IPv6), each protocol family flow has a unique Observation Domain ID. The following sections contain additional information:

Starting with Junos OS Release 17.3R1, IPFIX flow templates are supported on QFX10002 switches.

Starting with Junos OS Release 17.4R1, IPFIX flow templates are supported on QFX10008 and QFX10016 switches.

Starting with Junos OS Release 19.4R1, IPFIX flow templates are supported on SRX4100, SRX4200, SRX4600, SRX5400, SRX5600, SRX5800, vSRX, and vSRX3.0 devices.

Starting with Junos OS Release 20.1R1, IPFIX flow templates are supported on SRX300, SRX320, SRX340, SRX345, and SRX550HM devices.

Configuring the IPFIX Template Properties

To define the IPFIX templates, include the following statements at the **[edit services flow-monitoring version-ipfix]** hierarchy level:

```
[edit services flow-monitoring version-ipfix]
template template-name {
    options-template-id
    template-id
    observation-domain-id
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    option-refresh-rate packets packets seconds seconds;
    template-refresh-rate packets packets seconds seconds;
    (ipv4-template | ipv6-template);
}
```

The following details apply to the configuration statements:

- You assign each template a unique name by including the **template *template-name*** statement.
- You then specify each template for the appropriate type of traffic by including the **ipv4-template** or **ipv6-template**.
- Within the template definition, you can optionally include values for the **flow-active-timeout** and **flow-inactive-timeout** statements. These statements have specific default and range values when they are used in template definitions; the default is 60 seconds and the range is from 10 through 600 seconds.
- You can also include settings for the **option-refresh-rate** and **template-refresh-rate** statements within a template definition. For both of these properties, you can include a timer value (in seconds) or a packet count (in number of packets). For the **seconds** option, the default value is 600 and the range is from 10 through 600. For the **packets** option, the default value is 4800 and the range is from 1 through 480,000.

- To filter IPv6 traffic on a media interface, the following configuration is supported:

```
interfaces interface-name {
    unit 0 {
        family inet6 {
            sampling {
                input;
                output;
            }
        }
    }
}
```

Restrictions

The following restrictions apply to IPFIX templates:

- Outbound Routing Engine traffic is not sampled. A firewall filter is applied as output on the egress interface, which samples packets and exports the data. For transit traffic, egress sampling works correctly. For internal traffic, the next hop is installed in the Packet Forwarding Engine but sampled packets are not exported.
- Flows are created only after the route record resynchronization operation is complete, which takes 120 seconds.
- The VLAN ID field is updated when a new flow record is created and so, any change in VLAN ID after the record has been created might not be updated in the record.

Customizing Template ID, Observation Domain ID, and Source ID for IPFIX flow Templates

Starting in Junos OS Release 14.1, you can define an IPFIX flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic. Templates and the fields included in the template are transmitted to the collector periodically, and the collector need not be aware of the router configuration. You can specify the unique identifier for the version 9 and IPFIX templates. The identifier of a template is locally unique within a combination of a transport session and an observation domain. Template IDs 0 through 255 are reserved for template sets, options template sets, and other sets for future use. Template IDs of data sets are numbered from 256 through 65535. Typically, this information element or field in the template is used to define the characteristics or properties of other information elements in a template. After a restart of the export process of templates is performed, you can reassign template IDs.

This functionality to configure template ID, options template ID, observation domain ID, and source ID is supported on all routers with MPCs.

The corresponding data sets and option data sets contain the value of the template IDs and options template IDs respectively in the set ID field. This method enables the collector to match a data record with a template record.

For more information about specifying the source ID, observation domain ID, template ID, and options template ID for version 9 and IPFIX flows, see "Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows" on page 527 and "Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows" on page 531.

Fields Included in the IPv4 Template

- IPv4 Source Address
- IPv4 Destination Address
- IPv4 ToS
- IPv4 Protocol
- L4 Source Port
- L4 Destination Port
- ICMP Type and Code
- Input Interface
- VLAN ID
- IPv4 Source Mask
- IPv4 Destination Mask
- Source AS
- Destination AS
- IPv4 Next Hop Address
- TCP Flags
- Output Interface
- Number of Flow Bytes
- Number of Flow Packets
- Minimum TTL (time to live)
- Maximum TTL (time to live)
- Flow Start Time
- Flow End Time
- Flow End Reason

Fields Included in the IPv6 Template

- IPv6 Source Address
- IPv6 Destination Address
- IPv6 ToS
- IPv6 Protocol
- L4 Source Port
- L4 Destination Port
- ICMP Type and Code
- Input Interface
- VLAN ID
- 802.1Q VLAN identifier (dot1qVlanId)
- 802.1Q Customer VLAN identifier (dot1qCustomerVlanId)
- IPv6 Source Mask
- IPv6 Destination Mask
- Source AS
- Destination AS
- IPv6 Next Hop Address
- TCP Flags
- Output Interface
- Number of Flow Bytes
- Number of Flow Packets
- Minimum Hop Limits
- Maximum Hop Limits
- Flow Start Time
- Flow End Time
- Flow End Reason
- Fragment Identification (Starting in Junos OS Release 14.2)
- IPv6 Extension Headers (Starting in Junos OS Release 14.2)

Verification

The following show commands are supported for IPFIX:

- **show services accounting flow inline-jflow fpc-slot *fpc-slot***
- **show services accounting errors inline-jflow fpc-slot *fpc-slot***
- **show services accounting status inline-jflow fpc-slot *fpc-slot***

Example: Configuring IPFIX Flow Templates and Flow Sampling

The following example shows an IPFIX template configuration:

```
services {
    flow-monitoring {
        version-ipfix {
            template ipv4 {
                flow-active-timeout 60;
                flow-inactive-timeout 70;
                template-refresh-rate seconds 30;
                option-refresh-rate seconds 30;
                ipv4-template;
            }
        }
    }
}
```

```
chassis;
    fpc 0 {
        sampling-instance s1;
    }
```

The following example applies the IPFIX template to enable sampling of traffic for billing:

```
forwarding-options {
    sampling {
        instance {
            s1 {
                input {
                    rate 10;
                }
                family inet {
                    output {
```

```
                flow-server 192.0.2.2 {
                    port 2055;
                    version-ipfix {
                        template {
                            ipv4;
                        }
                    }
                }
                inline-jflow {
                    source-address 198.51.100.1;
                }
            }
        }
    }
  }
}
```

Release History Table

Release	Description
20.1R1	Starting with Junos OS Release 20.1R1, IPFIX flow templates are supported on SRX300, SRX320, SRX340, SRX345, and SRX550HM devices.
19.4R1	Starting with Junos OS Release 19.4R1, IPFIX flow templates are supported on SRX4100, SRX4200, SRX4600, SRX5400, SRX5600, SRX5800, vSRX, and vSRX3.0 devices.
17.4R1	Starting with Junos OS Release 17.4R1, IPFIX flow templates are supported on QFX10008 and QFX10016 switches.
17.2R1	Starting with Junos OS Release 17.3R1, IPFIX flow templates are supported on QFX10002 switches.
14.2	Fragment Identification (Starting in Junos OS Release 14.2)
14.2	IPv6 Extension Headers (Starting in Junos OS Release 14.2)
14.1	Starting in Junos OS Release 14.1, you can define an IPFIX flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic.

RELATED DOCUMENTATION

- Understanding Flow Aggregation | 488
- Including Fragmentation Identifier and IPv6 Extension Header Elements in IPFIX Templates on MX Series Routers | 536
- Enabling Flow Aggregation | 489
- Configuring Flow Aggregation on MX, M and T Series Routers and NFX250 to Use Version 5 or Version 8 cflowd | 490
- Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

Configuring Flow Aggregation to Use IPFIX Flow Templates on PTX Series Routers

IN THIS SECTION

Use of IPFIX allows you to define a flow record template suitable for IPv4 traffic or IPv6 traffic. Templates are transmitted to the collector periodically, and the collector is not aware of the router configuration. You can define template refresh rate, flow active timeout and inactive timeout.

If flow records are being sent for multiple protocol families (for example, for IPv4 and IPv6), each protocol family flow will have a unique Observation Domain ID.

The following sections contain additional information:

Configuring the IPFIX Template Properties

To define the IPFIX templates, include the following statements at the **[edit services flow-monitoring version-ipfix]** hierarchy level:

```
[edit services flow-monitoring IPFIX]
template name {
    options-template-id
    template-id
    observation-domain-id
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    option-refresh-rate packets packets seconds seconds;
    template-refresh-rate packets packets seconds seconds;
    (ipv4-template | ipv6-template);
}
```

The following details apply to the configuration statements:

- You assign each template a unique name by including the **template *name*** statement.
- You then specify each template for the appropriate type of traffic by including the **ipv4-template** or **ipv6–template**.
- Within the template definition, you can optionally include values for the **flow-active-timeout** and **flow-inactive-timeout** statements. These statements have specific default and range values when they are used in template definitions; the default is 60 seconds and the range is from 10 through 600 seconds.
- You can also include settings for the **option-refresh-rate** and **template-refresh-rate** statements within a template definition. For both of these properties, you can include a timer value (in seconds) or a packet count (in number of packets). For the **seconds** option, the default value is 600 and the range is from 10 through 600. For the **packets** option, the default value is 4800 and the range is from 1 through 480,000.
- To filter IPv6 traffic on a media interface, the following configuration is supported:

```
interfaces interface-name {
    unit 0 {
        family inet6 {
            sampling {
                input;
                output;
            }
        }
    }
}
```

Restrictions

The following restrictions apply to IPFIX templates:

- Outbound Routing Engine traffic is not sampled. A firewall filter is applied as output on the egress interface, which samples packets and exports the data. For transit traffic, egress sampling works correctly. For internal traffic, the next hop is installed in the Packet Forwarding Engine but sampled packets are not exported.
- Flows are created only after the route record resynchronization operation is complete, which takes 120 seconds.

Customizing Template ID, Observation Domain ID, and Source ID for IPFIX flow Templates

> **NOTE:** For PTX Series routers with third generation FPCs installed, the FPC's slot number is used for the observation domain ID.

Use of IPFIX flow templates allow you to define a flow record template suitable for IPv4 traffic, IPv6 traffic, MPLS traffic, a combination of IPv4 and MPLS traffic, or peer AS billing traffic. Templates and the fields included in the template are transmitted to the collector periodically, and the collector does not need to be aware of the router configuration. Template IDs 0 through 255 are reserved for template sets, options template sets, and other sets for future use. Template IDs of data sets are numbered from 256 through 65535. Typically, this information element or field in the template is used to define the characteristics or properties of other information elements in a template. After a restart of the export process of templates is performed, template IDs can be reassigned.

The corresponding data sets and option data sets contain the value of the template IDs and options template IDs respectively in the set ID field. This method enables the collector to match a data record with a template record.

Fields Included in the IPv4 Templates for PTX Series Routers

Table 88 on page 509 shows the fields that are available in the IPv4 templates.

Table 90: IPv4 Template Fields

Field	Element ID
IPv4 Source Address	8
IPv4 Destination Address	12

Table 90: IPv4 Template Fields *(continued)*

Field	Element ID
IPv4 TOS	5
IPv4 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code	32
Input Interface	10
Source AS	16
Destination AS	17
BGP Next Hop Address	18
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv4 Next Hop Address	15
IPv4 Source Mask	9
IPv4 Destination Mask	13
TCP Flags	6
IP Protocol Version	60
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153

Table 90: IPv4 Template Fields *(continued)*

Field	Element ID
The type of interface where packets are being received. This field can have the following values: • 1—Other (default value) • 131—De-encapsulated GRE traffic is reported as *tunnel*	368

Fields Included in the IPv6 Templates for PTX Series Routers

Table 89 on page 510 shows the fields that are available in the IPv6 templates.

Table 91: IPv6 Template Fields

Field	Element ID
IPv6 Source Address	27
IPv6 Destination Address	28
IPv6 TOS	5
IPv6 Protocol	4
L4 Source Port	7
L4 Destination Port	11
ICMP Type and Code (IPv6)	139
Input Interface	10
IPv6 Source Mask	29
IPv6 Destination Mask	30
TCP Flags	6
IP Protocol Version	60
Source AS	16
Destination AS	17

Table 91: IPv6 Template Fields *(continued)*

Field	Element ID
Output Interface	14
Number of Flow Bytes	1
Number of Flow Packets	2
Time the flow started with respect to system up time (FPC up time)	22
Time the flow ended with respect to system up time (FPC up time)	21
IPv6 Next Hop Address	62
Time the flow started with respect to Epoch time	152
Time the flow ended with respect to Epoch time	153
The type of interface where packets are being received. This field can have the following values: • 1—Other (default value) • 131—De-encapsulated GRE traffic is reported as *tunnel*	368

Verification

The following show commands are supported for IPFIX:

- **show services accounting flow inline-jflow fpc-slot *fpc-slot***
- **show services accounting errors inline-jflow fpc-slot *fpc-slot***
- **show services accounting status inline-jflow fpc-slot *fpc-slot***

Example: Configuring an IPFIX Flow Template and Flow Sampling

The following is a sample IPFIX template configuration:

```
services {
    flow-monitoring {
        version-ipfix {
            template ipv4 {
                flow-active-timeout 60;
```

```
            flow-inactive-timeout 70;
            template-refresh-rate seconds 30;
            option-refresh-rate seconds 30;
            ipv4-template;
        }
    }
  }
}
```

```
chassis;
  fpc 0 {
    sampling-instance s1;
  }
```

The following example applies the IPFIX template to enable sampling of traffic for billing:

```
forwarding-options {
  sampling {
    instance {
      s1 {
        input {
          rate 10;
        }
        family inet {
          output {
            flow-server 11.11.4.2 {
              port 2055;
              version-ipfix {
                template {
                  ipv4;
                }
              }
            }
            inline-jflow {
              source-address 11.11.2.1;
            }
          }
        }
      }
    }
  }
}
```

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring on PTX Series Routers | 458

version-ipfix | 1219

ipv4-template | 1013

ipv6-template | 1017

Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows

For IPFIX flows, an identifier of an Observation Domain is locally unique to an exporting process of the templates. The export process uses the Observation Domain ID to uniquely identify to the collection process in which the flows were metered. We recommend that you configure this ID to be unique for each IPFIX flow. A value of 0 indicates that no specific Observation Domain is identified by this information element. Typically, this attribute is used to limit the scope of other information elements. If the observation domain is not unique, the collector cannot uniquely identify an IPFIX device.

If you configure the same Observation Domain ID for different template types, such as for IPv4 and IPv6, it does not impact flow monitoring because the actual or the base observation domain ID is transmitted in the flow. The actual observation domain ID is derived from the value you configure and also in conjunction with other parameters such as the slot number, lookup chip (LU) instance, Packet Forwarding Engine instance. Such a method of computation of the observation domain ID ensures that this ID is not the same for two IPFIX devices.

Until Junos OS Release 13.3, the observation domain ID is predefined and is set to a fixed value, which is derived from the combination of FPC slot, sampling protocol, PFE Instance and LU Instance fields. This derivation creates a unique observation domain per LU per family. Starting with Junos OS Release 14.1, you can configure the observation domain ID, which causes the first 8 bits of the field to be configured.

The following modifications have been made:

- FPC slots are expanded to 8 bits to enable more slots to be configured in an MX Series Virtual Chassis configuration.
- 8 bits of the configured observation domain ID are used.
- You can configure a value for the observation domain ID in the range of 0 through 255.
- The Protocol field is increased to 3 bits to provide support for additional protocols in inline flow monitoring.
- You can associate the observation domain ID with templates by using the **observation-domain-id** ***domain-id*** statement at the **[edit services flow- monitoring version-ipfix template** ***template-name*****]** hierarchy level.

Starting with Junos OS Release 17.2R1, IPFIX flows are supported on QFX10002 switches.

Starting with Junos OS Release 17.4R1, IPFIX flows are supported on QFX10008 and QFX10016 switches.

For version 9 flows, a 32-bit value that identifies the Exporter Observation Domain is called the source ID. NetFlow collectors use the combination of the source IP address and the source ID field to separate different export streams originating from the same exporter.

To specify the observation domain ID for IPFIX flows, include the **observation-domain-id *domain-id*** statement at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.

```
[edit services flow-monitoring version-ipfix]
template template-name {
    observation-domain-id domain-id;
}
```

To specify the source ID for version 9 flows, include the **source-id *source-id*** statement at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level.

```
[edit services flow-monitoring version9]
template template-name {
    source-id source-id;
}
```

Table 92 on page 528 describes observation domain ID values for different combinations of the configured domain ID, protocol family, FPC slot, and the Packet Forwarding Engine and lookup chip instances.

Table 92: Example of Observation Domain ID

Configured Value	Protocol Family	FPC Slot	PFE Inst	LU Inst	Observation Domain Id Conf val rsvd 1proto slot LUInst PFEInst xxxx xxxx xxxx 1xxx xxxx xxxx xxxx xxxx
None	IPv4 (0)	1	1	0	0000 0000 0000 1000 0000 0001 0000 0001 0x00080101

Table 92: Example of Observation Domain ID *(continued)*

Configured Value	Protocol Family	FPC Slot	PFE Inst	LU Inst	Observation Domain Id Conf val rsvd 1proto slot LUInst PFEInst xxxx xxxx xxxx 1xxx xxxx xxxx xxxx xxxx
None	IPv6 (1)	1	1	0	0000 0000 0000 1001 0000 0001 0000 0001 0x00090101
None	VPLS (2)	1	1	0	0000 0000 0000 1010 0000 0001 0000 0001 0x000A0101
None	MPLS (3)	1	1	0	0000 0000 0000 1011 0000 0001 0000 0001 0x000B0101
4	IPv4 (0)	1	1	0	0000 0100 0000 1000 0000 0001 0000 0001 0x04080101
190	IPv4 (0)	1	1	0	1101 1110 0000 1000 0000 0001 0000 0001 0xBE080101
4	IPv4 (0)	2	1	1	0000 0100 0000 1000 0000 0010 0001 0001 0x04080211

Table 92: Example of Observation Domain ID *(continued)*

Configured Value	Protocol Family	FPC Slot	PFE Inst	LU Inst	Observation Domain Id Conf val rsvd 1proto slot LUInst PFEInst xxxx xxxx xxxx 1xxx xxxx xxxx xxxx xxxx
4	IPv6 (1)	1	1	0	0000 0100 0000 1001 0000 0001 0001 0000 0x04090110
190	IPv6 (1)	1	1	0	1101 1110 0000 1001 0000 0001 0001 0000 0xBE090110
4	VPLS (2)	2	2	0	0000 0100 0000 1010 0000 0010 0010 0000 0x040A0220
10	IPv4 (0)	28	2	1	0000 1010 0000 1000 0001 1100 0010 0001 0x0A081C21

Release History Table

Release	Description
17.4R1	Starting with Junos OS Release 17.4R1, IPFIX flows are supported on QFX10008 and QFX10016 switches.
17.2R1	Starting with Junos OS Release 17.2R1, IPFIX flows are supported on QFX10002 switches.

RELATED DOCUMENTATION

Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows | 531

Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows

Starting with Junos OS Release 14.1, you can define the template ID for version 9 and IPFIX templates for inline flow monitoring. To specify the template ID for version 9 flows, include the **template-id *id*** statement at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level.

```
[edit services flow-monitoring version9]
template template-name {
    template-id id;
}
```

To specify the template ID for version IPFIX flows, include the **template-id** statement at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.

```
[edit services flow-monitoring version-ipfix]
template template-name {
    template-id id;
}
```

To specify the options template ID for version 9 flows, include the **options-template-id** statement at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level.

```
[edit services flow-monitoring version9]
template template-name {
    options-template-id id;
}
```

To specify the options template ID for version IPFIX flows, include the **options-template-id** statement at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level. The template ID and options template ID can be a value in the range of 1024 through 65535.

```
[edit services flow-monitoring version-ipfix]
template template-name {
    options-template-id id;
}
```

Starting with Junos OS Release 17.2R1, IPFIX templates are supported on QFX10002 switches.

The template ID and options template ID can be a value in the range of 1024 through 65535. If you do not configure values for the template ID and options template ID, default values are assumed for these IDs, which are different for the various address families. If you configure the same template ID or options template ID value for different address families, such a setting is not processed properly and might cause unexpected behavior. For example, if you configure the same template ID value for both IPv4 and IPv6, the collector validates the export data based on the template ID value that it last receives. In this case, if IPv6 is configured after IPv4, the value is effective for IPv6 and the default value is used for IPv4.

The following are the default values of template IDs for IPFIX flows for the different protocols or address families, until Junos OS Release 13.3:

- IPv4 IPFIX flow template ID—256
- IPv6 IPFIX flow template ID—257
- VPLS IPFIX flow template ID—258
- MPLS IPFIX flow template ID—259

The following are the default values of template IDs for version 9 flows for the different protocols or address families, starting with Junos OS Release 14.1:

- IPv4 version 9 flow template ID—320
- IPv6 version 9 flow template ID—321
- VPLS version 9 flow template ID—322
- MPLS version 9 flow template ID—323

The following are the default values of template IDs for IPFIX flows for the different protocols or address families, until Junos OS Release 13.3:

- IPv4 IPFIX flow options template ID—512
- IPv6 IPFIX flow options template ID—513
- VPLS IPFIX flow options template ID—514
- MPLS IPFIX flow options template ID—515

The following are the default values of template IDs for version 9 flows for the different protocols or address families, starting with Junos OS Release 14.1:

- IPv4 version 9 flow options template ID—576
- IPv6 version 9 flow options template ID—577
- VPLS version 9 flow options template ID—578
- MPLS version 9 flow options template ID—579

Table 93 on page 533 describes the values of data template and option template IDs for different protocols with default and configured values for IPFIX flows.

Table 93: Values of Template and Option Template IDs for IPFIX Flows

Family	Configured Value	Data Template	Option Template
IPv4	None	256	576
IPv4	1024-65535	1024-65535	1024-65535
IPv6	None	257	577
IPv6	1024-65535	1024-65535	1024-65535
VPLS	None	258	578
VPLS	1024-65535	1024-65535	1024-65535
MPLS	None	259	579
MPLS	1024-65535	1024-65535	1024-65535

Table 94 on page 533 describes the values of data template and option template IDs for different protocols with default and configured values for version 9 flows.

Table 94: Values of Template and Option Template IDs for Version 9 Flows

Family	Configured Value	Data Template	Option Template
IPv4	None	320	576
IPv4	1024-65535	1024-65535	1024-65535
IPv6	None	321	577
IPv6	1024-65535	1024-65535	1024-65535
VPLS	None	322	578
VPLS	1024-65535	1024-65535	1024-65535
MPLS	None	323	579
MPLS	1024-65535	1024-65535	1024-65535

Table 95 on page 534 describes the values of data template and option template IDs for different protocols with default and configured values for IPFIX flows.

Table 95: Values of Template and Option Template IDs for IPFIX Flows

Configured Value	Protocol Family	FPC Slot	PFE Inst	LU Inst	Observation Domain Id Conf val rsvd 1proto slot LUInst PFEInst xxxx xxxx xxxx 1xxx xxxx xxxx xxxx xxxx
None	IPv4 (0)	1	1	0	0000 0000 0000 1000 0000 0001 0000 0001 0x00080101
None	IPv6 (1)	1	1	0	0000 0000 0000 1001 0000 0001 0000 0001 0x00090101
None	VPLS (2)	1	1	0	0000 0000 0000 1010 0000 0001 0000 0001 0x000A0101
None	MPLS (3)	1	1	0	0000 0000 0000 1011 0000 0001 0000 0001 0x000B0101
4	IPv4 (0)	1	1	0	0000 0100 0000 1000 0000 0001 0000 0001 0x04080101
190	IPv4 (0)	1	1	0	1101 1110 0000 1000 0000 0001 0000 0001 0xBE080101

Table 95: Values of Template and Option Template IDs for IPFIX Flows *(continued)*

Configured Value	Protocol Family	FPC Slot	PFE Inst	LU Inst	Observation Domain Id Conf val rsvd 1proto slot LUInst PFEInst xxxx xxxx xxxx 1xxx xxxx xxxx xxxx xxxx
4	IPv4 (0)	2	1	1	0000 0100 0000 1000 0000 0010 0001 0001 0x04080211
4	IPv6 (1)	1	1	0	0000 0100 0000 1001 0000 0001 0001 0000 0x04090110
190	IPv6 (1)	1	1	0	1101 1110 0000 1001 0000 0001 0001 0000 0xBE090110
4	VPLS (2)	2	2	0	0000 0100 0000 1010 0000 0010 0010 0000 0x040A0220
10	IPv4 (0)	28	2	1	0000 1010 0000 1000 0001 1100 0010 0001 0x0A081C21

Release History Table

Release	Description
17.2R1	Starting with Junos OS Release 17.2R1, IPFIX templates are supported on QFX10002 switches.

RELATED DOCUMENTATION

Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows | 527

Including Fragmentation Identifier and IPv6 Extension Header Elements in IPFIX Templates on MX Series Routers

Starting with Junos OS Release 14.2, the following attributes can be contained in IPFIX flow templates that are sent to the flow collector:

- fragmentIdentification (element ID 54)
- ipv6ExtensionHeaders (element ID 64)

A flow can receive many fragments in a given interval. For a given set of fragments of a packet, there is a unique fragment Identification. Hence, multiple such values can be received in a given interval. RFC 5102 for fragmentIdentification 54 does not clearly indicate which fragment identification needs to be shipped in the flow record information (first fragment observed after sending the flow record information or the last observed before shipping the flow record information). However, the last observed fragment Identification for a given flow is also transmitted to the flow collector.

Unlike in IPv4, IPv6 routers never fragment IPv6 packets. Packets exceeding the size of the maximum transmission unit of the destination link are dropped and this condition is signaled by a Packet Too Big ICMPv6 type 2 message to the originating node, similarly to the IPv4 method when the Don't Fragment (DF) bit is set.

The fragmentIdentification element is supported for both IPv4 and IPv6 flow templates. The fragmentIdentification element is added in the record template. The fragmentIdentification attribute is 32 bits in size for both IPv4 and IPv6. For IPv6, this field is present in fragment Extension header and Fragment Identifier is updated as 0 if there is no Fragment extension header.

Ports are a part of the key used to identify a Flow and the subsequent packets after the first fragmented packet does not have the port information. For a fragmented packet that is destined to the router, the packets that are split assume different flows (the first and the subsequent packets). Also, because the port is denoted as zeroes for fragmented packets, all the traffic destined to a particular destination from a particular source might be reported as the same flow, although no association exists between them in terms of destination ports. Fragment ID is not part of the key. Although the fragment ID attribute is unique between each source and destination, they might end up as same flows in the intermediate router.

With ports being used in the key for the flow lookup, the fragmented packets of a stream are accounted in two different flows. The first fragmented packet, which contains the port information in its packet, is part of one flow. Subsequent packets after the first fragments, which do not contain the port information, are accounted under a different flow. Because the second flow does not contain the port information to

identify itself, it consolidates all the other traffic streams with same source IP and destination IP address prefixes (also includes the non-first fragmented packets sent on different ports).

Destination nodes or endpoints in IPv6 are expected to perform path MTU discovery to determine the maximum size of packets to send, and the upper-layer protocol is expected to limit the payload size. However, if the upper-layer protocol is unable to do so, the sending host can use the Fragment extension header in order to perform end-to-end fragmentation of IPv6 packets. Any data link layer conveying IPv6 data must be capable of delivering an IP packet containing 1280 bytes without the need to invoke end-to-end fragmentation at the IP layer.

The ipv6ExtensionHeaders information element is a set for 32 bit fields. Each bit in this set represents one IPv6 Extension header. An extension header bit is set if that particular extension header is observed for the flow. The bit is set to 1 if any observed packet of this Flow contains the corresponding IPv6 extension header. Otherwise, if no observed packet of this Flow contained the respective IPv6 extension header, the value of the corresponding bit is 0. The ipv6ExtensionHeaders element is added in the record template. The number of flows that are created depends on the number of IPv6 packets that include the IPv6 extender header attribute.

To enable the inclusion of element ID, 54, fragmentIdentification and element ID, 64, ipv6ExtensionHeaders in IPFIX flow templates that are exported to the flow collector, include the **ipv6-extended-attrib** statement at the **[edit chassis fpc *slot-number* inline- services flow-table-size]** hierarchy level. Collection of IP4 fragmentation IDs occurs automatically without having to configure this setting explicitly.

```
[edit chassis]
fpc slot-number {
   inline-services {
      flow-table-size {
         ipv6-extended-attrib;
      }
   }
}
```

Starting in Junos OS Releases 17.3R4, 17.4R3, 18.1R4, 18.2R2, 18.3R2, and 18.4R1, the values of the IPv6 options and their functions that are contained in IPv6 packets are described in Table 96 on page 537.

Table 96: Values of IPv6 Options and Extension Headers in Packets

Bit Value	IPv6 Option	Next Header Code	Description
0	DST	60	Destination option header
1	HOP	0	Hop-by-hop option header
2	Res	Not applicable	Reserved

Table 96: Values of IPv6 Options and Extension Headers in Packets *(continued)*

Bit Value	IPv6 Option	Next Header Code	Description
3	UNK	Not applicable	Unknown layer 4 header (compressed, encrypted, not supported)
4	FRA0	44	Fragment header – first fragment
5	RH	43	Routing header
6	FRA1	44	Fragmentation header – not first fragment
7	Res	Not applicable	Reserved
8 through 11	Res	Not applicable	Reserved
12	MOB	135	IPv6 mobility (RFC3775)
13	ESP	50	Encrypted security payload
14	AH	51	Authentication header
15	PAY	108	Payload compression header
16 through 31	Res	Not applicable	Reserved

For Junos OS Releases prior to 17.3R4, 17.4R3, 18.1R4, 18.2R2, and 18.3R2, the values of the IPv6 options and their functions that are contained in IPv6 packets are described in Table 97 on page 538.

Table 97: Values of IPv6 Options and Extension Headers in Packets

Bit Value	IPv6 Option	Next Header Code	Description
0	Res	Not applicable	Reserved
1	FRA1	44	Fragmentation Header
2	RH	43	Routing Header

Table 97: Values of IPv6 Options and Extension Headers in Packets *(continued)*

Bit Value	IPv6 Option	Next Header Code	Description
3	FRA0	44	Fragment Header – First Fragment
4	UNK	Not applicable	Unknown Layer 4 header (compressed, encrypted, not supported)
5	Res	Not applicable	Reserved
6	HOP	0	Hop-by-hop option header
7	DST	60	Destination option header
8	PAY	108	Payload compression header
9	AH	51	Authentication header
10	ESP	50	Encrypted security payload
11 through 31	Res	Not applicable	Reserved

Release History Table

Release	Description
17.3R4	Starting in Junos OS Releases 17.3R4, 17.4R3, 18.1R4, 18.2R2, 18.3R2, and 18.4R1, the values of the IPv6 options and their functions that are contained in IPv6 packets are described in Table 96 on page 537.

RELATED DOCUMENTATION

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

ipv6-extended-attrib | **1016**

Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers

IN THIS SECTION

You can configure replication of the sampled flow records for use by multiple flow servers. You can use either sampling based on the Routing Engine, using cflowd version 5 or version 8, or sampling based on the services PIC, using flow aggregation version 9, as described in the following sections:

Directing Replicated Routing Engine–Based Sampling Flows to Multiple Servers

Routing Engine–based sampling supports up to eight flow servers for both cflowd version 5 and version 8 configurations. The total number of servers is limited to eight regardless of how many are configured for cflowd v5 or v8.

When you configure cflowd-based sampling, the export packets are replicated to all flow servers configured to receive them. If two servers are configured to receive v5 records, both the servers receive records for a specified flow.

> **NOTE:** With Routing Engine–based sampling, if multiple flow servers are configured with version 8 export format, all of them must use the same aggregation type. For example, all servers receiving version 8 export can be configured for **source-destination** aggregation type.

The following configuration example allows replication of export packets to two flow servers.

```
forwarding-options {
   sampling {
      instance inst1 {
         input {
            rate 1;
         }
         family inet;
            output {
```

```
            flow-server 10.10.3.2 {
              port 2055;
              version 5;
              source-address 192.168.164.119;
            }
            flow-server 172.17.20.62 {
              port 2055;
              version 5;
              source-address 192.168.164.119;
            }
          }
        }
      }
    }
}
```

Directing Replicated Version 9 Flow Aggregates to Multiple Servers

The export packets generated for a template are replicated to all the flow servers that are configured to receive information for that template. The maximum number of servers supported is eight.

This also implies that periodic updates required by version 9 (RFC 3954) are sent to each configured collector. The following updates are sent periodically as part of this requirement:

- Options data
- Template definition

The refresh period for options data and template definition is configured on a per-template basis at the **[edit services flow-monitoring]** hierarchy level.

The following configuration example allows replication of version 9 export packets to two flow servers.

```
forwarding-options {
  sampling {
    instance inst1 {
      input {
        rate 1;
      }
      family inet;
        output {
          flow-server 10.10.3.2 {
            port 2055;
            version9 {
```

```
                template {
                    ipv4;
                }
            }
        }
        flow-server 172.17.20.62 {
            port 2055;
            version9 {
                template {
                    ipv4;
                }
            }
        }
    }
    flow-inactive-timeout 30;
    flow-active-timeout 60;
    interface sp-4/0/0 {
        source-address 10.10.3.4;
    }
  }
 }
}
```

RELATED DOCUMENTATION

- Active Flow Monitoring Overview | 45
- Configuring Flow Monitoring | 3
- Configuring Services Interface Redundancy with Flow Monitoring | 61
- Example: Configuring Active Monitoring on an M, MX or T Series Router's Logical System | 48

Logging cflowd Flows on M and T Series Routers Before Export

To collect the cflowd flows in a log file before they are exported, include the **local-dump** statement at the **[edit forwarding-options sampling output flow-server *hostname*]** hierarchy level:

```
[edit forwarding-options sampling output flow-server hostname]
local-dump;
```

By default, the flows are collected in **/var/log/sampled**; to change the filename, include the **filename** statement at the **[edit forwarding-options sampling traceoptions]** hierarchy level. For more information about changing the filename, see "Configuring Traffic Sampling Output" on page 380.

> **NOTE:** Because the **local-dump** statement adds extra overhead, you should use it only while debugging cflowd problems, not during normal operation.

The following is an example of the flow information. The AS number exported is the origin AS number. All flows that belong under a cflowd header are dumped, followed by the header itself:

```
Jun 27 18:35:43 v5 flow entry
Jun 27 18:35:43    Src addr: 192.0.2.1
Jun 27 18:35:43    Dst addr: 198.51.100.15
Jun 27 18:35:43    Nhop addr: 198.51.100.240
Jun 27 18:35:43    Input interface: 5
Jun 27 18:35:43    Output interface: 3
Jun 27 18:35:43    Pkts in flow: 15
Jun 27 18:35:43    Bytes in flow: 600
Jun 27 18:35:43    Start time of flow: 7230
Jun 27 18:35:43    End time of flow: 7271
Jun 27 18:35:43    Src port: 26629
Jun 27 18:35:43    Dst port: 179
Jun 27 18:35:43    TCP flags: 0x10
Jun 27 18:35:43    IP proto num: 6
Jun 27 18:35:43    TOS: 0xc0
Jun 27 18:35:43    Src AS: 7018
Jun 27 18:35:43    Dst AS: 11111
Jun 27 18:35:43    Src netmask len: 16
Jun 27 18:35:43    Dst netmask len: 0
```

[... 41 more version 5 flow entries; then the following header:]

```
Jun 27 18:35:43 cflowd header:
Jun 27 18:35:43   Num-records: 42
Jun 27 18:35:43   Version: 5
Jun 27 18:35:43   low seq num: 118
Jun 27 18:35:43   Engine id: 0
Jun 27 18:35:43   Engine type: 3
```

RELATED DOCUMENTATION

Active Flow Monitoring Overview | 45
Configuring Flow Monitoring | 3
Directing Replicated Flows from M and T Series Routers to Multiple Flow Servers | 540
Configuring Services Interface Redundancy with Flow Monitoring | 61
Example: Configuring Active Monitoring on an M, MX or T Series Router's Logical System | 48

Configuring Next-Hop Address Learning on MX Series Routers for Destinations Accessible Over Multiple Paths

Starting in Junos OS Release 16.1, you can enable learning of next-hop addresses to correctly report the next hop address, output SNMP, destination IP address, and destination IP mask values in the flow records when a destination is reachable through multiple paths. By default, this behavior of learning the next-hop addresses is disabled for inline flow monitoring. When learning next-hop addresses is disabled, data is reported as follows:

- If the destination address of the sampled IPv4 flow is reachable through multiple paths, the IPv4 next hop address and the output SNMP address are reported in the flow records as the same as the gateway address and SNMP index of the first path seen in the forwarding table.
- If the destination address of the sampled IPv6 flow is reachable through multiple paths, the IPv4 next hop address and the output SNMP address are reported as 0 in the flow records.
- If the Incoming Interface (IIF) and Outgoing Interface (OIF) are not in the same VRF, then the destination IP address, destination IP mask, IPv4 next hop address, and the ouput SNMP address are reported as 0 in the flow records.

When learning of next-hop addresses is enabled, output SNMP, destination IP address, and destination IP mask values in the flow records are reported correctly when a destination is reachable through multiple paths.. To enable next-hop learning, include the **nexthop-learning enable** statement at the **[edit services flow-monitoring (version-ipfix | version9) template *template-name*]** hierarchy level.

```
[edit services flow-monitoring (version-ipfix | version9) template template-name]
set nexthop-learning enable;
```

Release History Table

Release	Description
16.1	Starting in Junos OS Release 16.1, you can enable learning of next-hop addresses to correctly report the next hop address, output SNMP, destination IP address, and destination IP mask values in the flow records when a destination is reachable through multiple paths.

RELATED DOCUMENTATION

nexthop-learning | **1054**

CHAPTER 13

Sending Packets for Analysis Using Port Mirroring

IN THIS CHAPTER

Understanding Port Mirroring

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.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

Example: Configuring Multiple Port Mirroring with Next-Hop Groups on M, MX and T Series Routers | 172

Rerouting Packets on an M, MX or T Series Router with Port Mirroring

You can copy packets and reroute them to another interface by using port mirroring. To send packet copies to an interface, include the **interface** statement at the **[edit forwarding-options port-mirroring** family ***family-name*** **output]** hierarchy level and specify the interface to receive the traffic.

You can even send port-mirrored traffic to a monitoring services or adaptive services interface. If you choose this option, accepted traffic is copied and the packet copies are sent to the services interface for flow processing.

To configure how often packets are copied from the monitored traffic, include the **rate** statement at the **[edit forwarding-options port-mirroring** family ***family-name*** **input]** hierarchy level. A rate of **1** port-mirrors every packet, while a rate of **10** port-mirrors every tenth packet.

```
[edit]
forwarding-options {
    port-mirroring {
        family (inet | inet6) {
            input {
                rate 1;
            }
            output {
                interface sp-2/0/0.0;
            }
        }
    }
}
```

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers

IN THIS SECTION

To prepare traffic for port mirroring, include the **filter** statement at the **[edit firewall family inet]** hierarchy level:

```
filter filter-name;
```

This filter at the **[edit firewall family (inet | inet6)]** hierarchy level selects traffic to be port-mirrored:

```
filter filter-name {
    term term-name {
        then {
            port-mirror;
            accept;
        }
    }
}
```

To configure port mirroring on a logical interface, configure the following statements at the **[edit forwarding-options port-mirroring]** hierarchy level:

```
[edit forwarding-options port-mirroring]
input {
    maximum-packet-length bytes;
    rate rate;
    run-length number;
}
family (inet | inet6) {
    output {
        interface interface-name {
            next-hop address;
        }
        next-hop-group group-name {
            group-type (inet | inet6);
            interface interface-name {
                next-hop address;
            }
            next-hop-subgroup group-name {
                interface interface-name {
                    next-hop address;
                }
            }
        }
        no-filter-check;
    }
```

```
}
```

NOTE: The PTX series does not support egress port mirroring.

The ACX6360 router does not support egress port mirroring.

Specify the port-mirroring destination by including the **next-hop** statement at the **[edit forwarding-options port-mirroring family (inet | inet6) output interface *interface-name*]** hierarchy level:

```
next-hop address;
```

NOTE: For IPv4 port mirroring to reach a next-hop destination, you must manually include a static Address Resolution Protocol (ARP) entry in the router configuration.

You can also specify the port-mirroring destination by including the **next-hop-group** statement at the **[edit forwarding-options port-mirroring family (inet | inet6) output]** hierarchy level. Starting in Junos OS Release 14.2R1, the **next-hop-group** statement for the port-mirroring destination is supported for inet6.

NOTE: The ACX6360 router does not support the **next-hop**, **next-hop-group**, and the **maximum-packet-length**statements.

```
next-hop-group group-name {
    group-type (inet | inet6);
    interface interface-name {
        next-hop address;
    }
    next-hop-subgroup group-name {
        interface interface-name {
            next-hop address;
        }
    }
}
```

The **no-filter-check** statement is required when you send port-mirrored traffic to a Tunnel PIC that has a filter applied to it.

The interface used to send the packets to the analyzer is the output interface configured above at the **[edit forwarding-options port-mirroring family (inet | inet6) output]** hierarchy level. You can use any physical interface type, including generic routing encapsulation (GRE) tunnel interfaces. The next-hop address specifies the destination address; this statement is mandatory for non point-to-point interfaces, such as Ethernet interfaces.

To configure the sampling rate or duration, include the **rate** or **run-length** statement at the **[edit forwarding-options port-mirroring input]** hierarchy level.

You can trace port-mirroring operations the same way you trace sampling operations. For more information, see "Tracing Traffic Sampling Operations" on page 383.

For more information about port mirroring, see the following sections:

Configuring Tunnels

In typical applications, you send the sampled packets to an analyzer or a workstation for analysis, rather than another router. If you must send this traffic over a network, you should use tunnels. For more information about tunnel interfaces, see *Tunnel Services Overview*.

The MX Series routers support Dense Port Concentrators (DPCs) with built-in Ethernet ports, which do not support Tunnel Services PICs. To create tunnel interfaces on an MX Series router with a DPC, you configure the DPC and the corresponding Packet Forwarding Engine to use for tunneling services at the **[edit chassis]** hierarchy level. You also configure the amount of bandwidth reserved for tunnel services. The Junos OS creates tunnel interfaces on the Packet Forwarding Engine.

To create tunnel interfaces on MX Series routers, include the following statements at the **[edit chassis]** hierarchy level:

```
[edit chassis]
fpc slot-number {
    pic number {
        tunnel-services {
            bandwidth bandwidth-value;
        }
    }
}
```

Include the **fpc *slot-number*** statement to specify the slot number of the DPC. If two SCBs are installed, the range is 0 through . If three SCBs are installed, the range is 0 through 5 and 7 through .

Include the **pic *number*** statement to specify the number of the Packet Forwarding Engine on the DPC. The range is 0 through 3.

You can also specify the amount of bandwidth to allocate for tunnel traffic on each Packet Forwarding Engine by including the **bandwidth** *bandwidth-value* statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

- **1g** indicates that 1 Gbps of bandwidth is reserved for tunnel traffic. Configure this option only for a Packet Forwarding Engine on a Gigabit Ethernet 40-port DPC.
- **10g** indicates that 10 Gbps of bandwidth is reserved for tunnel traffic. Configure this option only for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.
- **20g** or **40g**—Configure 20 gigabits per second or 40 gigabits per second only on an MX Series router with the MPC3E and the 100-Gigabit CFP MIC.

If you specify a bandwidth that is not compatible with the type of DPC and Packet Forwarding Engine, tunnel services are not activated. For example, you cannot specify a bandwidth of 1 Gbps for a Packet Forwarding Engine on a 10-Gigabit Ethernet 4-port DPC.

When you configure tunnel interfaces on the Packet Forwarding Engine of a 10-Gigabit Ethernet 4-port DPC, the Ethernet interfaces for that port are removed from service and are no longer visible in the command-line interface (CLI). The Packet Forwarding Engine of a 10-Gigabit Ethernet 4-port DPC supports either tunnel interfaces or Ethernet interfaces, but not both. Each port on the 10-Gigabit Ethernet 4-port DPC includes two LEDs, one for tunnel services and one for Ethernet services, to indicate which type of service is being used. On the Gigabit Ethernet 40-port DPC, you can configure both tunnel and Ethernet interfaces at the same time.

If your router is equipped with a Tunnel PIC, you can forward duplicate packets to multiple interfaces by configuring a next-hop group. To configure a next-hop group, include the **next-hop-group** statement at the **[edit forwarding-options]** hierarchy level:

```
[edit forwarding-options]
next-hop-group group-names {
    interface interface-name {
        next-hop address;
    }
}
```

The **interface** statement specifies the interface that sends out sampled information. The **next-hop** statement specifies the next-hop addresses to which to send the sampled information.

Starting in Junos OS Release 14.2, for IPv6 port mirroring to reach next-hop destination, you can configure a **next-hop-group** statement at the **[edit forwarding-options port-mirroring family inet6 output]** hierarchy level:

```
next-hop-group group-name{
    group-type inet6;
```

```
    interface interface-name {
      next-hop ipv6-address;
    }
    next-hop-subgroup group-name{
      interface interface-name {
        next-hop ipv6-address;
      }
    }
}
```

Next-hop groups have the following restrictions:

- Next-hop groups are supported for inet, inet6, and bridge family.
- Next-hop groups are supported on M Series and MX Series routers.
- Next-hop groups or next-hop subgroups support up to 16 next-hop addresses.
- Up to 30 next-hop groups are supported.
- Each next-hop group is expected to have at least two next-hop addresses.
- Each next-hop subgroup supports up to 16 next-hop groups.

Port Mirroring with Next-Hop Groups

You can configure next-hop groups for M Series, MX Series, and TX Series routers using either IP addresses or Layer 2 addresses for the next hops. Use the **group-type [inet | inet6 | layer-2]** statement at **[edit forwarding-options next-hop-group *next-hop-group-name*]** hierarchy level to establish the next-hop groups. (The **inet6** option is available starting in Junos OS Release 14.2.) You can reference more than one port mirroring instance in a filter on MX Series routers. Use the **port-mirror-instance *instance-name*** statement at the **[edit firewall family *family-name* filter *filter-name* term *term-name*]** hierarchy level to refer to one of several port mirroring instances.

> **NOTE:** On MX Series routers with MPCs, port mirroring instances can only be bound to the FPC level and not up to the PIC level. For MX Series routers with a DPC card, both levels are supported.

On M Series, MX Series, and T Series routers only, you can configure port mirroring using next-hop groups, also known as *multipacket port mirroring*, without the presence of a Tunnel PIC. To configure this functionality, include the **next-hop-group** statement at the **[edit forwarding-options port-mirror family [inet | inet6] output]** (the **inet6** option is available starting in Junos OS Release 14.2.) or **[edit forwarding-options port-mirror instance *instance-name* family inet output]** hierarchy level:

```
[edit forwarding-options]
port-mirror {
    family inet {
        output {
            next-hop-group group-name {
                interface interface-name {
                    next-hop address;
                }
            }
        }
    }
}
```

or

```
[edit forwarding-options]
port-mirror {
    family inet6 {
        output {
            next-hop-group group-name{
                group-type inet6;
                interface interface-name {
                    next-hop ipv6-address;
                }
                next-hop-subgroup group-name{
                    interface interface-name {
                        next-hop ipv6-address;
                    }
                }
            }
        }
    }
}
```

or

```
[edit forwarding-options]
port-mirror {
    instance instance-name {
        family (inet | vpls) {
            output {
                next-hop-group group-name;
            }
```

```
        }
    }
}
```

You define the next-hop group by including the **next-hop-group** statement at the **[edit forwarding-options]** hierarchy level. For an example, see "Examples: Configuring Port Mirroring" on page 557. This configuration is supported with IPv4 and IPv6 addresses.

You can disable this configuration by including a **disable** or **disable-all-instances** statement at the **[edit forwarding-options port-mirror]** hierarchy level or by including a **disable** statement at the **[edit forwarding-options port-mirror instance *instance-name*]** hierarchy level. You can display the settings and network status by issuing the **show forwarding-options next-hop-group** and **show forwarding-options port-mirroring** operational commands.

NOTE: If you try to bind any derived instance to the FPC, a commit error occur.

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's **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 MX Series routers with MPCs.

Using inline port mirroring, a port-mirror instance 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 192.0.2.3;
            }
        }
    }
}
```

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 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 is sampled using the input parameters specified by the referred instance but the copy is sent to the its own destination.

Filter-Based Forwarding with Multiple Monitoring Interfaces

If port-mirrored packets are to be distributed to multiple monitoring or collection interfaces based on patterns in packet headers, it is helpful to configure a filter-based forwarding (FBF) filter on the port-mirroring egress interface.

When an FBF filter is installed as an output filter, a packet that is forwarded to the filter has already undergone at least one route lookup. After the packet is classified at the egress interface by the FBF filter, it is redirected to another routing table for additional route lookup. Obviously, the route lookup in the latter routing table (designated by an FBF routing instance) must result in a different next hop from those from the previous tables the packet has passed through, to avoid packet looping inside the Packet Forwarding Engine.

For more information about FBF configuration, see the *Junos OS Routing Protocols Library*. For an example of FBF applied to an output interface, see "Examples: Configuring Port Mirroring" on page 557.

Restrictions

The following restrictions apply to port-mirroring configurations:

- The interface you configure for port mirroring should not participate in any kind of routing activity.
- The destination address you specify should not have a route to the ultimate traffic destination. For example, if the sampled IPv4 packets have a destination address of **10.68.9.10** and the port-mirrored traffic is sent to **10.68.20.15** for analysis, the device associated with the latter address should not know a route to **10.68.9.10**. Also, it should not send the sampled packets back to the source address.
- IPv4 and IPv6 traffic is supported. For IPv6 port mirroring, you must configure the next-hop router with an IPv6 neighbor before mirroring the traffic, similar to an ARP request for IPv4 traffic. All the restrictions applied to IPv4 configurations should also apply to IPv6.
- On M120 and M320 Series routers, multiple next-hop mirroring is not supported.
- Because M320 Series routers do not support multiple bindings of port-mirror instances per FPC, the **port-mirror-instance** action is not supported in firewall filters for these routers.
- Port mirroring in the ingress and egress direction is not supported for link services IQ (lsq-) interfaces.
- The PTX platform does not support egress port mirroring.

- On M Series routers other than the M120 and M320 Series routers, only one family protocol (either IPv4 or IPv6) is supported at a time.
- Port mirroring supports up to 16 next hops.
- Only transit data is supported.
- You can configure multiple port-mirroring interfaces per router.
- On routers containing an Internet Processor II application-specific integrated circuit (ASIC), you must include a firewall filter with both the **accept** action and the **port-mirror** action modifier on the inbound interface. Do not include the **discard** action, or port mirroring does not work.
- If the port-mirroring interface is a non-point-to-point interface, you must include an IP address under the **port-mirroring** statement to identify the other end of the link. This IP address must be reachable for you to see the sampled traffic. If the port-mirroring interface is an Ethernet interface, the router should have an Address Resolution Protocol (ARP) entry for it. The following sample configuration sets up a static ARP entry.
- You do not need to configure firewall filters on both inbound and outbound interfaces, but at least one is necessary on the inbound interface to provide the copies of the packets to send to an analyzer.
- Inline port mirroring is supported only on MX Series routers with MPCs.
- Configuration for both port mirroring and traffic sampling are handled by the same daemon, so in order to view a trace log file for port mirroring, you must configure the **traceoptions** option under traffic sampling.

Configuring Port Mirroring on Services Interfaces

A special situation arises when you configure unit **0** of a services interface (AS or Multiservices PIC) to be the port-mirroring logical interface, as in the following example:

```
[edit forwarding-options]
port-mirroring {
    input {
        rate 1;
    }
    family inet {
        output {
            interface sp-1/0/0.0;
        }
    }
}
```

Since any traffic directed to unit **0** on a services interface is targeted for monitoring (cflowd packets are generated for it), the sample port-mirroring configuration indicates that the customer wants to have cflowd records generated for the port-mirrored traffic.

However, generation of cflowd records requires the following additional configuration; if it is missing, the port-mirrored traffic is simply dropped by the services interface without generating any cflowd packets.

```
[edit forwarding-options]
sampling {
    instance instance1 { # named instances of sampling parameters
        input {
            rate 1;
        }
        family inet {
            output {
                flow-server 172.16.28.65 {
                    port 1230;
                }
                interface sp-1/0/0 {  # If the port-mirrored traffic requires monitoring, this
                                      # interface must be same as that specified in the
                                       # port-mirroring configuration.
                    source-address 198.51.100.3;
                }
            }
        }
    }
}
```

NOTE: Another way to configure sp-1/0/0 to generate cflowd records is to use only the sampling configuration, but include a firewall filter **sample** action instead of a **port-mirror** action.

Examples: Configuring Port Mirroring

The following example sends port-mirrored traffic to multiple cflowd servers or packet analyzers:

```
[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 10.11.0.1/24;
        }
      }
}
ge-1/1/0 { # This is an exit interface for HTTP packets.
    unit 0 {
        family inet {
            address 10.12.0.1/24;
        }
    }
}
ge-1/2/0 { # This is an exit interface for HTTP packets.
    unit 0 {
        family inet {
            address 10.13.0.1/24;
        }
    }
}
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 10.2.2.2/30;
        }
    }
}
so-7/0/0 { # This is an exit interface for all remaining packets.
    unit 0 {
        family inet {
            address 10.5.5.5/30;
        }
    }
}
so-7/0/1 { # This is an exit interface for all remaining packets.
    unit 0 {
        family inet {
```

```
            address 10.6.6.6/30;
        }
    }
}
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.
            }
        }
    }
}
[edit forwarding-options]
port-mirroring { # This is required when you configure next-hop groups.
    input {
        rate 1; # This rate port mirrors one packet for every one received (1:1 = all
            # packets).
    }
    family inet {
        output { # This sends traffic to a tunnel interface to prepare for 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
        # name only.
    interface so-4/3/0.0;
    interface so-0/3/0.0;
}
next-hop-group http-traffic { # You need to configure a next hop for multipoint interfaces
        # (Ethernet).
    interface ge-1/1/0.0 {
        next-hop 10.12.0.2;
    }
    interface ge-1/2/0.0 {
        next-hop 10.13.0.2;
    }
}
next-hop-group default-collect {
```

```
    interface so-7/0/0.0;
    interface so-7/0/1.0;
}
[edit 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.
                accept;
            }
        }
    }
    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 term sends all remaining traffic to a final next-hop group.
            then next-hop-group default-collectors;
        }
    }
}
```

The following example demonstrates configuration of filter-based forwarding at the output interface. In this example, the packet flow follows this path:

1. A packet arrives at interface fe-1/2/0.0 with source and destination addresses **10.50.200.1** and **10.50.100.1**, respectively.
2. The route lookup in routing table **inet.0** points to the egress interface so-0/0/3.0.
3. The output filter installed at so-0/0/3.0 redirects the packet to routing table **fbf.inet.0**.
4. The packet matches the entry **10.50.100.0/25**, and finally leaves the router from interface so-2/0/0.0.

```
[edit interfaces]
```

```
so-0/0/3 {
    unit 0 {
        family inet {
            filter {
                output fbf;
            }
            address 10.50.10.2/25;
        }
    }
}
fe-1/2/0 {
    unit 0 {
        family inet {
            address 10.50.50.2/25;
        }
    }
}
so-2/0/0 {
    unit 0 {
        family inet {
            address 10.50.20.2/25;
        }
    }
}
[edit firewall]
filter fbf {
    term 0 {
        from {
            source-address {
                10.50.200.0/25;
            }
        }
        then routing-instance fbf;
    }
    term d {
        then count d;
    }
}
[edit routing-instances]
fbf {
    instance-type forwarding;
    routing-options {
        static {
            route 10.50.100.0/25 next-hop so-2/0/0.0;
```

```
        }
      }
    }
    [edit routing-options]
    interface-routes {
      rib-group inet fbf-group;
    }
    static {
      route 10.50.100.0/25 next-hop 10.50.10.1;
    }
    rib-groups {
      fbf-group {
        import-rib [ inet.0 fbf.inet.0 ];
      }
    }
```

The following example shows configuration of port mirroring using next-hop groups or multipacket port mirroring:

```
forwarding-options {
  next-hop-group inet_nhg {
    group-type inet;
    interface ge-2/0/2.101 {
      next-hop 10.2.0.2;
    }
    interface ge-2/2/8.2 {
      next-hop 10.8.0.2;
    }
  }
  next-hop-group vpls_nhg {
    group-type layer-2;
    interface ge-2/0/1.100;
    interface ge-2/2/9.0;
    inactive: next-hop-subgroup vpls_subg {
      interface ge-2/0/1.101;
      interface ge-2/2/9.1;
    }
  }
  next-hop-group vpls_nhg_2 {
    group-type layer-2;
    interface ge-2/2/1.100;
    interface ge-2/3/9.0;
  }
  port-mirror {
```

```
disable-all-instances; /* Disable all port-mirroring instances */
disable; /* Disable the global instance */
input {
   rate 10; # start mirroring every 10th packet
   run-length 4; # mirror 4 additional packets
}
family inet {
   output {
      next-hop-group inet_nhg;
   }
}
family inet6 {
   output {
      next-hop-group inet6_nhg6 {
         group-type inet6;
         interface ge-2/0/3.102 {
            next-hop 2001:db8::1:10 ;
         }
         interface ge-2/0/4.103 {
            next-hop 2001:db8::20:10;
         }
         next-hop-subgroup vpls_subg {
            interface ge-2/0/.101 {
               next-hop 2001:db8::3:1;
            }
            interface ge-2/2/9.1 {
               next-hop 2001:db8::4:1;
            }
         }
      }
   }
}
family vpls {
   output {
      next-hop-group vpls_nhg;
   }
}
instance {
   inst1 {
      disable; /* Disable this instance */
      input {
         rate 1;
         maximum-packet-length 200;
      }
```

```
            family inet {
               output {
                  next-hop-group inet_nhg;
               }
            }
            family inet6 {
               output {
                  next-hop-group inet6_nhg6;
               }
            }
            family vpls {
               output {
                  next-hop-group vpls_nhg_2;
               }
            }
         }
      }
   }
}
```

The following example shows configuration of port mirroring using next-hop groups or multipacket port mirroring on a T Series router:

```
forwarding-options {
   next-hop-group inet_nhg {
      group-type inet;
      interface so-0/0/0.0; # There is no need for the nexthop address on T Series routers
      interface ge-2/0/2/.0 {
         next-hop 203.0.113.4
      }
      next-hop-subgroup sub_inet {
         interface so-1/2/0.0;
         interface ge-6/1/2.0 {
            next-hop 203.0.113.137;
         }
      }
      next-hop-group vpls_nhg_2 {
         group-type layer-2;
         interface ge-2/2/1.100;
         interface ge-2/3/9.0;
      }
   }
   port-mirroring {
      disable-all-instances; /*Disable all port-mirroring instances */
```

```
        disable; /* Disable the global instance */
        input {
            rate 10;
            run-length 4;
        }
        family inet {
            output {
                next-hop-group inet_nhg;
            }
        }
        family vpls {
            output {
                next-hop-group vpls_nhg;
            }
        }
        instance {
            inst1 {
                disable; /* Disable this instance */
                input {
                    rate 1;
                    maximum-packet-length 200;
                }
                family inet {
                    output {
                        next-hop-group inet_nhg;
                    }
                }
                family vpls {
                    output {
                        next-hop-group vpls_nhg_2;
                    }
                }
            }
        }
    }
}
```

The following example shows configuration of inline port mirroring using PM1, PM2, and PM3 as our port mirror instances.

```
instance {
    pm1 {
        input {
            rate 3;
```

```
    }
    family inet {
        output {
            interface ge-1/2/2.0 {
                next-hop 192.0.2.130;
            }
        }
    }
}
pm2 {
    input-parameters-instance pm1;
    family inet {
        output {
            interface ge-1/2/3.0 {
                next-hop 192.0.2.3;
            }
        }
    }
}
pm3 {
    input {
        rate 3;
    }
    family inet6 {
        output {
            interface ge-1/2/3.0 {
                next-hop 2001:db8::5:1;
            }
        }
    }
}
firewall {
    filter pm_filter {
        term t1 {
            then port-mirror-instance pm2;
        }
    }
    filter nhg6_filter6 {
        term t6 {
            then next-hop-group inet6-nhg6;
        }
    }
}
chassis {
```

```
        fpc 1 {
            port-mirror-instance pm1;
        }
    }
```

The packets be sampled at a rate of 3, and the copy is sent to 192.0.2.3.

Release History Table

Release	Description
14.2R1	Starting in Junos OS Release 14.2R1, the **next-hop-group** statement for the port-mirroring destination is supported for inet6.
14.2	Starting in Junos OS Release 14.2, for IPv6 port mirroring to reach next-hop destination, you can configure a **next-hop-group** statement at the **[edit forwarding-options port-mirroring family inet6 output]** hierarchy level:
14.2	The **inet6** option is available starting in Junos OS Release 14.2.
14.2	the **inet6** option is available starting in Junos OS Release 14.2.

RELATED DOCUMENTATION

Understanding Port Mirroring | **546**

Example: Configuring Multiple Port Mirroring with Next-Hop Groups on M, MX and T Series Routers | **172**

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 includeport mirroringmultiple interfaces used to forward duplicate packets used in port mirroring.

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 additional 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 interface (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;
        }
    }
}
...
```

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.

RELATED DOCUMENTATION

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 47: Active Flow Monitoring—Multiple Port Mirroring with Next-Hop Groups Topology Diagram

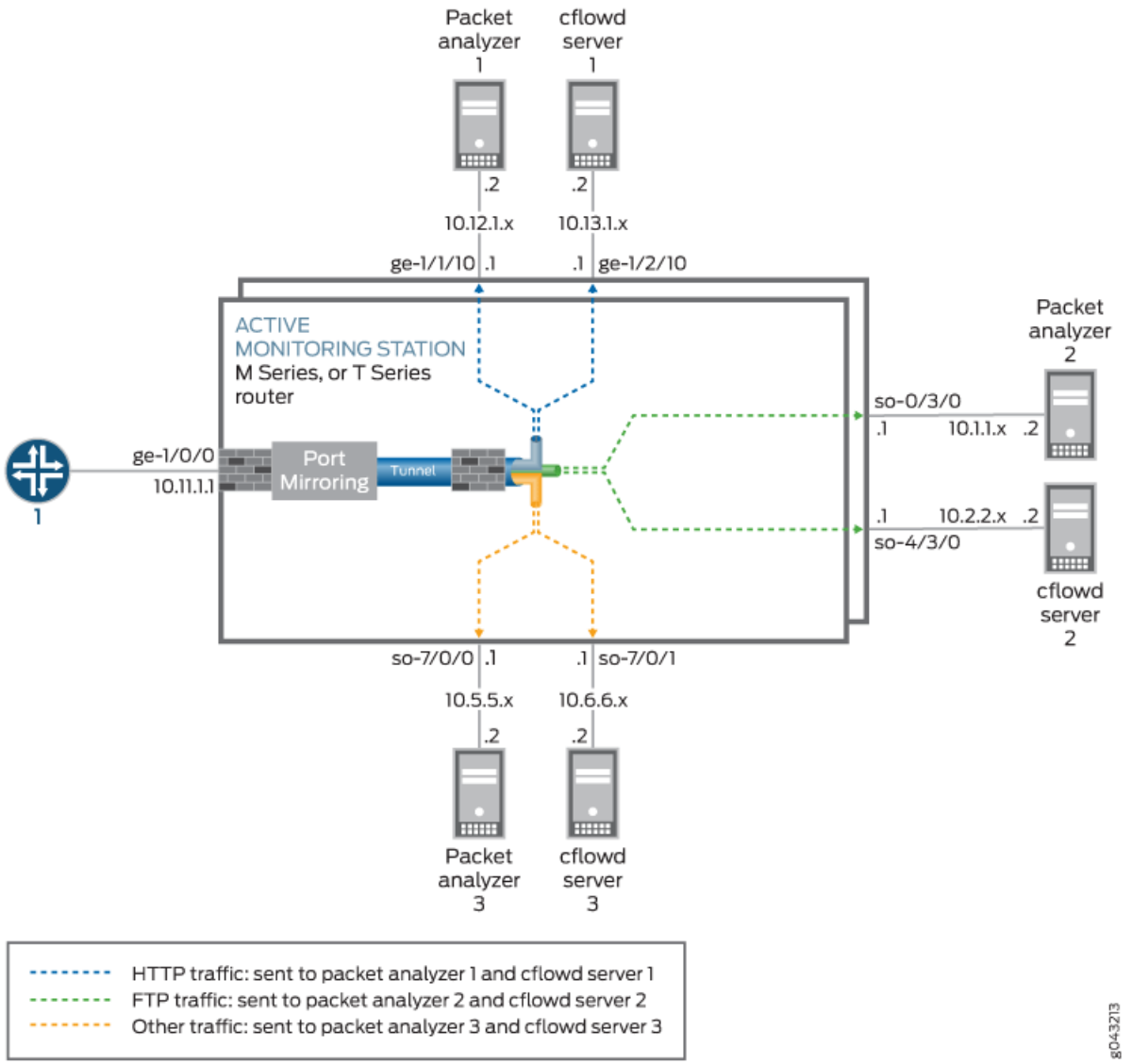


Figure 21 on page 173 shows an example of how to configure multiple port mirroring with next-hop groups. All traffic enters the monitoring router at interface ge-1/0/0. A firewall filter counts and port-mirrors all incoming packets to a Tunnel Services PIC. A second filter is applied to the tunnel interface 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;
            }
        }
    }
```

```
}
```

RELATED DOCUMENTATION

Understanding Port Mirroring | 546

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

5

PART

Real-Time Performance Monitoring and Video Monitoring Services

CHAPTER 14

Monitoring Traffic Using Real-Time Performance Monitoring

IN THIS CHAPTER

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers

Real-time performance monitoring (RPM) enables you to configure active probes to track and monitor traffic. Probes collect packets per destination and per application, including PING Internet Control Message Protocol (ICMP) packets, User Datagram Protocol and Transmission Control Protocol (UDP/TCP) packets with user-configured ports, user-configured Differentiated Services code point (DSCP) type-of-service (ToS) packets, and Hypertext Transfer Protocol (HTTP) packets. RPM provides Management Information Base (MIB) support with extensions for RFC 2925, *Definitions of Managed Objects for Remote Ping, Traceroute, and Lookup Operations*.

Starting in Junos OS Release 17.3R1, you can apply RPM to IPsec tunnels and GRE tunnels for PIC-based and Routing Engine-based RPM clients and servers if you are using MS-MPCs or MS-MICs. Packet Forwarding Engine-based RPM is not supported for IPsec tunnels. Support of RPM on IPSec tunnels enables service level agreement (SLA) monitoring for traffic transported in IPSec tunnels.

Starting in Junos OS Releases 19.1R1, PTX Series routers support timestamping of RPM probe messages on the Packet Forwarding Engine.

NOTE: RPM is not supported on logical systems.

NOTE: RPM is not supported when you enable Next Gen Services on an MX Series router.

You can also configure RPM services to determine automatically whether a path exists between a host router and its configured BGP neighbors. You can view the results of the discovery using an SNMP client. Results are stored in **pingResultsTable**, **jnxPingResultsTable**, **jnxPingProbeHistoryTable**, and **pingProbeHistoryTable**.

Probe configuration and probe results are supported by the command-line interface (CLI) and SNMP.

The following probe types are supported with DSCP marking:

- HTTP get (not available for BGP RPM services)
- ICMP echo
- ICMP timestamp
- TCP connection
- UDP echo
- UDP timestamp

With probes, you can monitor:

- Average round-trip time
- Jitter of the round-trip time—The difference between the minimum and maximum round-trip time
- Maximum round-trip time
- Minimum round-trip time
- Standard deviation of the round-trip time

One-way measurements for ICMP timestamp probes include:

- Minimum, maximum, standard deviation, and jitter measurements for egress and ingress times
- Number of probe responses received
- Number of probes sent
- Percentage of lost probes

You can configure the following RPM thresholds:

- Ingress/egress delay
- Jitter
- Round-trip time
- Standard deviation
- Successive lost probes
- Total lost probes (per test)

Support is also implemented for user-configured CoS classifiers and for prioritization of RPM packets over regular data packets received on an input interface.

Table 98 on page 580 provides information about RPM and related timestamp support on MPC, MS-MIC/MPC, and Routing Engine:

Table 98: RPM and related timestamp support

Feature	Role	IP Version	Support (Y/N)	Timestamp on Routing Engine	Timestamp on MPC (hardware-timestamp)	Timestamp on MPC (si-interface)	Timestamp on MS-MIC/MPC (delegate-probes)
RPM	Client	IPv4	Y	Y (µsec) 2000 maximum probes	Y (µsec) 2000 maximum probes	N	Y (msec) 1 million maximum probes
		IPv6	Y	Y (µsec) 2000 maximum probes	N	N	Y (msec) 1 million maximum probes
	Server	IPv4	Y	Y (µsec) 2000 maximum probes	Y (µsec) 2000 maximum probes	N	Y (msec) 1 million maximum probes
		IPv6	Y	Y (µsec) 2000 maximum probes	N	N	Y (msec) 1 million maximum probes

Release History Table

Release	Description
19.3R2	RPM is not supported when you enable Next Gen Services on an MX Series router.
19.1	Starting in Junos OS Releases 19.1R1, PTX Series routers support timestamping of RPM probe messages on the Packet Forwarding Engine.
17.3R1	Starting in Junos OS Release 17.3R1, you can apply RPM to IPsec tunnels and GRE tunnels for PIC-based and Routing Engine-based RPM clients and servers if you are using MS-MPCs or MS-MICs.

RELATED DOCUMENTATION

Real-Time Performance Monitoring on ACX Series

Real-time performance monitoring (RPM) allows the user to perform service-level monitoring. When RPM is configured on a router, the router calculates network performance based on packet response time, jitter, and packet loss. RPM is supported on all ACX Series routers. You can configure these values to be gathered by HTTP, Internet Control Message Protocol (ICMP), TCP, and UDP requests. The router gathers RPM statistics by sending out probes to a specified probe target, identified by an IP address. When the target receives a probe, it generates responses that are received by the router. You set the probe options in the **test *test-name*** statement at the [**edit services rpm probe *owner***] hierarchy level. You use the **show services rpm probe-results** command to view the results of the most recent RPM probes.

> **NOTE:** Packet Forwarding Engine timestamping is available only for ICMP probes and for UDP probes with the destination port set to UDP_ECHO port (7).

On ACX Series routers, the following statements are supported at the [**edit services rpm**] hierarchy level:

```
probe owner {
    test test-name {
        data-fill data;
        data-size size;
        destination-interface interface-name;
        destination-port port;
        dscp-code-point dscp-bits;
        hardware-timestamp;
        history-size size;
        moving-average-size number;
        one-way-hardware-timestamp;
        probe-count count;
        probe-interval seconds;
        probe-type type;
        routing-instance instance-name;
        source-address address;
        target (url url | address address);
        test-interval interval;
        thresholds thresholds;
        traps traps;
```

```
    }
}
```

NOTE: The ACX Series routers do not support the configuration of RPM probes to a routing instance along with the configuration of the **hardware-timestamp** statement.

NOTE: ACX5000 line of routers do not support **hardware-timestamp** feature for RPM.

RELATED DOCUMENTATION

Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers | **635**

show services rpm probe-results | **1357**

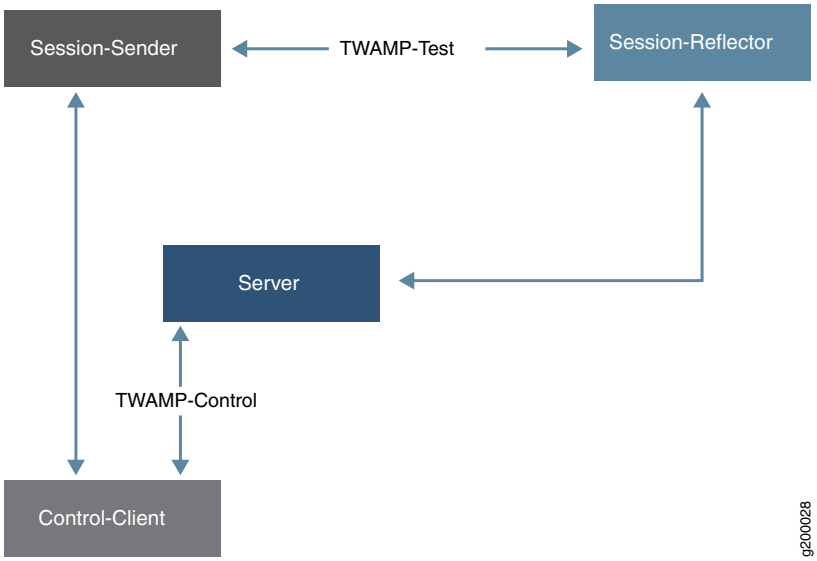
Understanding Two-Way Active Measurement Protocol on Routers

The Two-Way Active Management Protocol (TWAMP), described in RFC 5357, is an extension of the One-Way Active Management Protocol (OWAMP) that supplies two-way or round-trip measurements instead of unidirectional capabilities. Two-way measurements are helpful because round-trip delays do not require host clock synchronization and remote support might be a simple echo function. However, the Internet Control Message Protocol (ICMP) Echo Request/Reply (used by ping) for this purpose has several shortcomings. TWAMP defines an open protocol for measuring two-way or round-trip metrics with greater accuracy than other methods by using time-stamps (processing delays can be factored as well).

Usually, TWAMP operates between interfaces on two devices playing specific roles. TWAMP is often used to check Service Level Agreement (SLA) compliance, and the TWAMP feature is often presented in that context. TWAMP uses two related protocols, running between several defined entities: TWAMP-Control—Initiates, starts, and ends test sessions. The TWAMP-Control protocol runs between a Control-Client and a TWAMP Server. TWAMP-Test—Exchanges test packets between two TWAMP entities. The TWAMP-Test protocol runs between a Session-Sender and a Session-Reflector.

The four elements are shown in Figure 48 on page 583

Figure 48: Four Elements of TWAMP



Although four different TWAMP devices can perform the four logical roles of TWAMP Control-Client, Server, Session-Sender, and Session-Reflector, different devices can play different roles. A common implementation combines the roles of Control-Client and Session-Sender in one device (known as the TWAMP controller or TWAMP client) and the roles of Server and Session-Reflector in the other device (known as the TWAMP responder or TWAMP server). In this case, each device runs both the TWAMP-Control (between Control-Client and Server) and TWAMP-Test (between Session-Sender and Session-Reflector) protocols.

The TWAMP client-server architecture as implemented looks like this:

- TWAMP client
  - Control-Client sets up, starts and stops the TWAMP test sessions.
  - Session-Sender creates TWAMP test packets sent to the Session-Reflector in TWAMP server.
- TWAMP server
  - Session-Reflector sends back a measurement packet when a test packet is received, but does not maintain a record of such information.
  - Server manages one or more sessions with the TWAMP client and listens for control messages on a TCP port.

The packaging of these elements into TWAMP client and TWAMP server processes is shown in Figure 49 on page 584.

Figure 49: The Elements of TWAMP Implemented as Client (Left) and Server (Right).

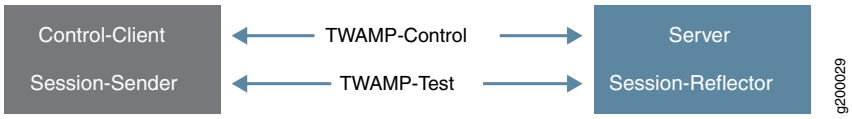


Table 99 on page 584 provides information about TWAMP and related timestamp support on MPC, MS-MIC/MPC, and Routing Engine:

Table 99: TWAMP and related timestamp support

Feature	Role	IP Version	Support (Y/N)	Timestamp on Routing Engine	Timestamp on MPC (hardware-timestamp)	Timestamp on MPC (si-interface)	Timestamp on MS-MIC/MPC (delegate-probes)
TWAMP	Client	IPv4	Y	N	Y (µsec) 500 maximum probes	Y (µsec) 500 maximum probes	N
		IPv6	N	N	N	N	N
	Server	IPv4	Y	N	Y (µsec) 500 maximum probes	Y (µsec) 500 maximum probes	N
		IPv6	N	N	N	N	N

TWAMP on MX Series routers

For Junos OS release 15.1, both the control client and session sender (the TWAMP client) reside on the same Juniper Networks router. However, the TWAMP client does not require that the server and the session reflector to be on the same system. Therefore, the Juniper TWAMP client is capable of working with a third-party server implementation.

NOTE: TWAMP is not supported when you enable Next Gen Services on an MX Series router.

TWAMP on PTX Series routers

Starting in Junos OS Release 19.2R1, the Two-Way Active Measurement Protocol (TWAMP) is supported on PTX series routers. The TWAMP-Control protocol is used to set up performance measurement sessions between a TWAMP client and a TWAMP server, and the TWAMP-Test protocol is used to send and receive performance measurement probes. The destination interface **si-x/y/z** attribute, which is meant for enabling inline services is not supported on PTX series routers for TWAMP client configurations.

TWAMP on ACX Series routers

The Two-Way Active Measurement Protocol (TWAMP) defines a standard for measuring IP performance between two devices in a network. You can configure TWAMP on ACX Series routers. ACX Series routers support only the reflector side of TWAMP.

For more information about TWAMP, see RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)*.

To configure TWAMP properties, include the **twamp** statement at the **[edit services rpm]** hierarchy level:

```
[edit services rpm]
twamp {
   server {
      authentication-mode mode;
      client-list list-name {
         [ address address ];
         authentication none];
      }
      max-connection-duration hours;
      maximum-connections count;
      maximum-connections-per-client count;
      maximum-sessions count;
      maximum-sessions-per-connection count;
      port udp-port-number;
      server-inactivity-timeout minutes;
   }
}
```

You can specify a number of TWAMP server properties, some of which are optional, by including the **server** statement at the **[edit services rpm twamp]** hierarchy level:

```
[edit services rpm twamp]
server {
   client-list list-name {
      [ address address ];
   }
```

```
    authentication-mode mode;
    maximum-connections count;
    maximum-connections-per-client count;
    maximum-sessions count;
    maximum-sessions-per-connection count;
    port udp-port-number;
}
```

- To specify the list of allowed control client hosts that can connect to this server, include the **client-list** statement at the **[edit services rpm twamp server]** hierarchy level. Each value you include must be a Classless Interdomain Routing (CIDR) address (IP address plus mask) that represents a network of allowed hosts. You can include multiple client lists, each of which can contain a maximum of 64 entries. You must configure at least one client address to enable TWAMP.
- ACX Series routers do not support authentication and encryption modes. The value for **authentication-mode** statement at the **[edit services rpm twamp server]** hierarchy level must be set to **none**.
- To specify the maximum number of concurrent connections the server can have to client hosts, include the **maximum-connections** statement at the **[edit services rpm twamp server]** hierarchy level. The allowed range of values is 1 through 500 and the default value is 64. You can also limit the number of connections the server can make to a particular client host by including the **maximum-connections-per-client** statement. ACX Series routers supports a maximum of 15 concurrent connections.
- To specify the maximum number of sessions the server can have running at one time, include the **maximum-sessions** statement at the **[edit services rpm twamp server]** hierarchy level. The allowed range of values is 1 through 500 and the default value is 64. You can also limit the number of sessions the server can have on a single connection by including the **maximum-sessions-per-connection** statement. ACX Series routers supports a maximum of 15 sessions.
- To specify the TWAMP server listening port, include the **port** statement at the **[edit services rpm twamp server]** hierarchy level. The range is 1 through 65,535. If a port range is not specified, then the default port 862 is used.
- TWAMP control connection traffic always arrives on ACX routers with the listening port set as 862. Because this port number for traffic probes can be modified, probes that arrive with a different port number are not recognized and processed by ACX routers correctly. As a result, TWAMP traffic and host-bound packets are dropped in such a scenario.

Release History Table

Release	Description
19.3R2	TWAMP is not supported when you enable Next Gen Services on an MX Series router.
19.2R1	Starting in Junos OS Release 19.2R1, the Two-Way Active Measurement Protocol (TWAMP) is supported on PTX series routers.

RELATED DOCUMENTATION

Example: Configuring TWAMP Client and Server on MX Series Routers | **621**

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | **610**

post-cli-implicit-firewall | **1087**

Understanding TWAMP Auto-Restart

After a network outage or a configuration change, when the Two-Way Active Management Protocol (TWAMP) client goes down, you have to manually start the TWAMP session by using **request services rpm twamp start client** command. Starting in Junos OS Release 19.1R1, the TWAMP client restarts automatically without any manual intervention.

The automatic restart of the TWAMP session enables the TWAMP client to initiate the TCP control connection and UDP test sessions automatically during the following scenarios:

- Immediately after the TWAMP client configuration is committed.
- After the remote operation daemon (rmopd) is started with the valid TWAMP client configuration presence.
- After the TWAMP client configuration is activated.
- Immediately after the TWAMP server is reachable from the TWAMP client, based on the **test-interval**.

When the network fails or the TWMP server becomes unreachable for any reason, the, TWAMP client tries to reconnect to the TWAMP server after every **test-interval** value until it is successful. However, for the client to reconnect to the TWAMP server automatically, the *test-count* value in the **set rpm twamp client control-connection *test-count*** command must be 0. At the TWAMP server side, the default value of ***max-connection-duration*** in the **set rpm twamp server max-connection-duration** must also be 0. Thereby, you can retain the connection until it is cleared.

> **NOTE:** Starting in Junos OS Release 19.1R1, the default value of *test-count* at the TWAMP client and *max-connection-duration* at the TWAMP server is 0.

After you configure and commit a TWAMP test, the client runs tests indefinitely—that is, it continues to send probes after the configured test interval even after a test is completed, and even if there is a network or server failure. You can stop the automatic running of tests by changing the value of the ***test-count*** option to a nonzero value. If you do that, the automatic restart feature is disabled, and you need to manually start the TWAMP client for it to establish connection with the server and start test sessions.

You can maintain and view the statistics related to the previous probes sent during server unavailability. You can Use the **set services rpm twamp client control-connection c1 persistent-results** command to preserve and display the test results after the network recovers or when the TWAMP server is again reachable.

Benefits

- You do not need to restart the TWAMP session manually after the client goes down as a result of a network outage or configuration change.
- You do not need to run an event script to restart TWAMP session from client side.

TCP Keepalive Support for TWAMP Client and Server

Keepalive probes can assert client (peers) when another peer becomes unreachable. If the problem is in the network between two peers, the keepalive action is to wait for some time and then retry sending the keepalive packet before marking the connection as broken.

When the keepalive timer for a TCP connection reaches zero, TCP client sends its peer a keepalive probe packet with no data in it and with the ACK flag turned on. The client receives a reply from the remote host with no data and with the ACK flag set. If the client receives a reply to its keepalive probe, the client can assert that the connection is still up and running. If the peer does not reply to the keepalive probe, you can assert that the connection cannot be considered valid and then take corrective action.

To detect the TWAMP control connection failures at TWAMP client and TWAMP servers, you need to configure the following parameters:

- **tcp-keepcnt**—Number of unacknowledged probes to send before considering the connection dead and notifying the application layer.
- **tcp-keepidle**—Time interval between the last data packet sent and the first keepalive probe sent.
- **tcp-keepintvl**—Time interval between successive keepalive probes.

Release History Table

Release	Description
19.1R1	Starting in Junos OS Release 19.1R1, the TWAMP client restarts automatically without any manual intervention.

RELATED DOCUMENTATION

- tcp-keepcnt | 1157
- tcp-keepintvl | 1159
- tcp-keepidle | 1158
- Configuring TWAMP Client and TWAMP Server to Reconnect Automatically After TWAMP Server Unavailability | 614

Two-Way Active Measurement Protocol on ACX Series

The Two-Way Active Measurement Protocol (TWAMP) defines a standard for measuring IP performance between two devices in a network. You can configure TWAMP on ACX Series routers. ACX Series routers support only the reflector side of TWAMP.

For more information about TWAMP, see RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)*.

To configure TWAMP properties, include the **twamp** statement at the **[edit services rpm]** hierarchy level:

```
[edit services rpm]
twamp {
   server {
      authentication-mode mode;
      client-list list-name {
         [ address address ];
         authentication none];
      }
      max-connection-duration hours;
      maximum-connections count;
      maximum-connections-per-client count;
      maximum-sessions count;
      maximum-sessions-per-connection count;
      port udp-port-number;
      server-inactivity-timeout minutes;
```

```
    }
}
```

You can specify a number of TWAMP server properties, some of which are optional, by including the **server** statement at the **[edit services rpm twamp]** hierarchy level:

```
[edit services rpm twamp]
server {
    client-list list-name {
        [ address address ];
    }
    authentication-mode mode;
    maximum-connections count;
    maximum-connections-per-client count;
    maximum-sessions count;
    maximum-sessions-per-connection count;
    port udp-port-number;
}
```

- To specify the list of allowed control client hosts that can connect to this server, include the **client-list** statement at the **[edit services rpm twamp server]** hierarchy level. Each value you include must be a Classless Interdomain Routing (CIDR) address (IP address plus mask) that represents a network of allowed hosts. You can include multiple client lists, each of which can contain a maximum of 64 entries. You must configure at least one client address to enable TWAMP.
- ACX Series routers do not support authentication and encryption modes. The value for **authentication-mode** statement at the **[edit services rpm twamp server]** hierarchy level must be set to **none**.
- To specify the maximum number of concurrent connections the server can have to client hosts, include the **maximum-connections** statement at the **[edit services rpm twamp server]** hierarchy level. The allowed range of values is 1 through 2048 and the default value is 64. You can also limit the number of connections the server can make to a particular client host by including the **maximum-connections-per-client** statement. ACX Series routers supports a maximum of 15 concurrent connections.
- To specify the maximum number of sessions the server can have running at one time, include the **maximum-sessions** statement at the **[edit services rpm twamp server]** hierarchy level. The allowed range of values is 1 through 2048 and the default value is 64. You can also limit the number of sessions the server can have on a single connection by including the **maximum-sessions-per-connection** statement. ACX Series routers supports a maximum of 15 sessions.

- To specify the TWAMP server listening port, include the **port** statement at the **[edit services rpm twamp server]** hierarchy level. The range is 1 through 65,535. If a port range is not specified, then the default port 862 is used.
- TWAMP control connection traffic always arrives on ACX routers with the listening port set as 862. Because this port number for traffic probes can be modified, probes that arrive with a different port number are not recognized and processed by ACX routers correctly. As a result, TWAMP traffic and host-bound packets are dropped in such a scenario.

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches

The probe owner and test name of an RPM probe together represent a single RPM configuration instance. When you specify the test name, you also can configure the test parameters.

To configure the probe owner, test name, and test parameters, include the **probe** statement at the **[edit services rpm]** hierarchy level:

```
[edit services rpm]
probe owner {
    delegate-probes;
    test test-name {
        data-fill data;
        data-size size;
        destination-interface interface-name;
        destination-port port;
        dscp-code-point dscp-bits;
        hardware-timestamp;
        history-size size;
        inet6-options;
        moving-average-size number;
        one-way-hardware-timestamp;
        probe-count count;
        probe-interval seconds;
        probe-type type;
        routing-instance instance-name;
        rpm-scale {
            destination {
                interface interface-name.logical-unit-number;
                subunit-cnt subunit-cnt;
            }
            source {
```

```
            address-base ipv4-address-base;
            count ipv4-count;
            step ipv4-step;
        }
        source-inet6 {
            address-base ipv6-address-base;
            count ipv6-count;
            step ipv6-step;
        }
        target {
            address-base ipv4-address-base;
            count ipv4-count;
            step ipv4-step;
        }
        target-inet6 {
            address-base ipv6-address-base;
            count ipv6-count;
            step ipv6-step;
        }
        tests-count tests-count;
    }
    source-address address;
    target (url url | address address);
    test-interval interval;
    thresholds thresholds;
    traps traps;
    ttl [hop-count]
  }
}
```

Keep the following points in mind when you configure RPM clients and RPM servers:

- RPM is not supported on logical systems.
- You cannot configure an RPM client that is PIC-based and an RPM server that is based on either the Packet Forwarding Engine or Routing Engine to receive the RPM probes.
- You cannot configure an RPM client that is Packet Forwarding Engine-based and an RPM server that receives the RPM probes to be on the PIC or Routing Engine.
- The RPM client and RPM server must be located on the same type of module. For example, if the RPM client is PIC-based, the RPM server must also be PIC-based, and if the RPM server is Packet Forwarding Engine-based, the RPM client must also be Packet Forwarding Engine-based.
- Starting in Junos OS Release 17.3R1, PIC-based and Routing Engine-based RPM is supported for IPsec tunnels and GRE tunnels if you are using MS-MPCs or MS-MICs. Packet Forwarding Engine-based RPM

is not supported for IPsec tunnels. Support of RPM on IPSec tunnels enables service level agreement (SLA) monitoring for traffic transported in IPSec tunnels.

- Starting in Junos OS Release 17.3R1, you can configure the generation of IPv4 **icmp-ping** and **icmp-ping-timestamp** RPM probes on an MS-MPC or MS-MIC, which increases the number of probes generated upto 1 million per second on every service-NPU compared to the number of probes that are generated on the Packet Forwarding Engine. Starting in Junos OS Release 18.1R1, you can configure the generation of **icmp6-ping** RPM probes on an MS-MPC or MS-MIC. To configure the generation of RPM probes on an MS-MPC or MS-MIC:

  - Include the **destination-interface** ***interface-name.logical-unit-number*** at the **[edit services rpm probe** ***owner*** **test** ***test-name*****]** hierarchy level, and include the **delegate-probes** statement at the **[edit services rpm probe** ***owner*****]** hierarchy level. The *interface-name.logical-unit-number* specifies a logical interface on an MS-MPC or MS-MIC slot, PIC, and port that has a valid IP address defined on it (for example, ms-1/2/1.1). The interface cannot be an aggregated multiservices interface (ams-).
  - Include the **rpm client-delegate-probes** and the **family (inet | inet6) address** ***address*** statements at the **[edit interfaces** ***interface-name*** **unit** ***logical-unit-number*****]** hierarchy level. The *interface-name* and the *logical-unit-number* must match the *interface-name.logical-unit-number* that you used for the **destination-interface**.

  For RPM probes configured on an MS-MPC or MS-MIC, you cannot configure the **routing-instance** statement at the **[edit services rpm probe** ***owner*** **test** ***test-name*****]** hierarchy level, and you cannot configure both IPv4 and IPv6 probes within the same test.

  Starting in Junos OS Release 18.1R1, you can use additional filters to limit the output of the **show services rpm probe-results** and **show services rpm history-results** commands for RPM probes generated on an MS-MPC or MS-MIC.

- Starting in Junos OS Release 17.4R1, you can optimize the CLI configuration for RPM tests for IPv4. Starting in Junos OS Release 18.2R1, you can also optimize the CLI configuration for RPM tests for IPv6. This optimization allows the use of minimal RPM configuration statements to generate multiple tests (up to 100K tests) with pre-defined, reserved RPM test names. This optimization can be configured for tests with probes that are generated by either the Packet Forwarding Engine or by an MS-MPC or MS-MIC. Tests are generated for multiple combinations of source and target addresses, which are incremented based on your configuration.

  The maximum number of concurrent RPM probes supported for various Junos releases are as follows:

  - Junos OS release older than 17.3R1—500
  - Junos OS release 17.3R1 and later—2000 for ICMP and ICMP-Timestamp probe types. For probes of other types (UDP and TCP) the limit is 500.
  - Junos OS Release 17.3R1 and later (with the implementation of **delegate-probes**)—1 Million per Service-NPU.

NOTE: One MS-MIC contains one service-NPU and one MS-MPC contains four service-NPUs.

With the implementation of **delegate-probes**, the RPM probes are compliant to RFC792 and RFC4443. Hence, they can be used to monitor any IP device compliant to either RFC and are able to respond to icmp-timestamp and/or icmp6-ping packets.

Tests are first generated for all the source addresses with the initial target address, then tests are generated for all the source addresses with the next available target address, and so on. You can also configure a group that contains global values for a particular probe owner, and apply the group to the probe owner.

To generate multiple RPM tests, configure the following:

```
[edit services rpm probe owner]
apply-groups group-name;
test test-name {
  rpm-scale {
    destination {
      interface interface-name.logical-unit-number;
      subunit-cnt subunit-cnt;
    }
    source {
      address-base ipv4-address-base;
      count ipv4-count;
      step ipv4-step;
    }
    source-inet6 {
      address-base ipv6-address-base;
      count ipv6-count;
      step ipv6-step;
    }
    target {
      address-base ipv4-address-base;
      count ipv4-count;
      step ipv4-step;
    }
    target-inet6 {
      address-base ipv6-address-base;
      count ipv6-count;
      step ipv6-step;
    }
    tests-count tests-count;
  }
```

```
}
```

The options are:

ipv4-address-base—The IPv4 source or target address that is incremented to generate the addresses used in the RPM tests.

ipv6-address-base—The IPv6 source or target address that is incremented to generate the addresses used in the RPM tests.

ipv4-step—The amount to increment the IPv4 source or target address for each generated RPM test.

ipv6-step—The amount to increment the IPv6 source or target address for each generated RPM test.

ipv4-count—The maximum number of IPv4 source or target addresses to use for the generated RPM tests.

ipv6-count—The maximum number of IPv6 source or target addresses to use for the generated RPM tests.

interface-name.logical-unit-number—The services interface that is generating RPM probes and the logical unit number that is used for the first test that is generated.

subunit-cnt—The maximum number of logical units used by the services interface in the generated tests. The first generated test uses the logical unit specified in the *interface-name.logical-unit-number* option, and each successive test increments the logical unit number by one. Once the maximum number of logical units has been used, the next generated test cycles back to the logical unit that was used in the first test.

tests-count—The maximum number of RPM tests to generate. This number must be less than or equal to the number of generated source addresses multiplied by the number of generated target addresses.

To configure a group with global values for a particular probe owner:

```
[edit groups group-name]
services {
   rpm {
      probe <*> {
         test {
            data-fill data;
            data-size size;
            dscp-code-point dscp-bits;
            history-size size;
            moving-average-size number;
            probe-count count;
            probe-type type;
```

```
            test-interval interval;
            thresholds thresholds;
        }
    }
  }
}
```

- To specify a probe owner, include the **probe** statement at the **[edit services rpm]** hierarchy level. The probe owner identifier can be up to 32 characters in length.
- To specify a test name, include the **test** statement at the **[edit services rpm probe *owner*]** hierarchy level. The test name identifier can be up to 32 characters in length. A test represents the range of probes over which the standard deviation, average, and jitter are calculated.
- To specify the contents of the data portion of Internet Control Message Protocol (ICMP) probes, include the **data-fill** statement at the **[edit services rpm probe *owner*]** hierarchy level. The value can be a hexadecimal value. The **data-fill** statement is not valid with the **http-get** or **http-metadata-get** probe types.
- To specify the size of the data portion of ICMP probes, include the **data-size** statement at the **[edit services rpm probe *owner*]** hierarchy level. The size can be from **0** through **65400** and the default size is **0**. The **data-size** statement is not valid with the **http-get** or **http-metadata-get** probe types.

  NOTE: If you configure the hardware timestamp feature (see "Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches" on page 602):

  - This is a deprecated element **data-size** default value is 32 bytes and this is a deprecated element 32 is the minimum value for explicit configuration. The UDP timestamp probe type is an exception; it requires a minimum data size of 44 bytes.
  - The **data-size** must be at least 100 bytes smaller than the default MTU of the interface of the RPM client interface.

- On M Series and T Series routers, you configure the **destination-interface** statement to enable hardware timestamping of RPM probe packets. You specify an sp- interface to have the AS or Multiservices PIC add the hardware timestamps; for more information, see "Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches" on page 602. You can also include the **one-way-hardware-timestamp** statement to enable one-way delay and jitter measurements.
- To specify the User Datagram Protocol (UDP) port or Transmission Control Protocol (TCP) port to which the probe is sent, include the **destination-port** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. The **destination-port** statement is used only for the UDP and TCP probe types. The value can be **7** or from **49160** through **65535**.

When you configure either **probe-type udp-ping** or **probe-type udp-ping-timestamp** along with hardware timestamping, the value for the **destination-port** can be only 7. A constraint check prevents you from configuring any other value for the destination port in this case. This constraint does not apply when you are using one-way hardware timestamping.

- To specify the value of the Differentiated Services (DiffServ) field within the IP header, include the **dscp-code-point** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. The DiffServ code point (DSCP) bits value can be set to a valid 6-bit pattern; for example, **001111**. It also can be set using an alias configured at the **[edit class-of-service code-point-aliases dscp]** hierarchy level. The default is **000000**.
- To specify the number of stored history entries, include the **history-size** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. Specify a value from **0** to **512**. The default is **50**.
- To specify a number of samples for making statistical calculations, include the **moving-average-size** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. Specify a value from **0** through **255**.
- To specify the number of probes within a test, include the **probe-count** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. Specify a value from **1** through **15**.
- To specify the time to wait between sending packets, include the **probe-interval** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. Specify a value from **1** through **255** seconds.
- To specify the packet and protocol contents of the probe, include the **probe-type** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. The following probe types are supported:
  - **http-get**—Sends a Hypertext Transfer Protocol (HTTP) get request to a target URL.
  - **http-metadata-get**—Sends an HTTP get request for metadata to a target URL.
  - **icmp-ping**—Sends ICMP echo requests to a target address.
  - **icmp-ping-timestamp**—Sends ICMP timestamp requests to a target address.
  - **tcp-ping**—Sends TCP packets to a target.
  - **udp-ping**—Sends UDP packets to a target.
  - **udp-ping-timestamp**—Sends UDP timestamp requests to a target address.

  The following probe types support hardware timestamping of probe packets: **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, **udp-ping-timestamp**. Starting in Junos OS Release 17.3R3, the delegate probes are distributed evenly across the interval of 3 seconds to avoid the packet bursts in the network due to real-time performance monitoring (RPM). RPM syslogs are processed with the increase in the ramp up time of RPM delegates tests to 60 seconds. With RPM syslogs processed, the chances of multiple tests starting and ending at the same time are smaller, thus a potential restriction in **event-processing**.

> **NOTE:** Some probe types require additional parameters to be configured. For example, when you specify the **tcp-ping** or **udp-ping** option, you must configure the destination port using the **destination-port** statement. The **udp-ping-timestamp** option requires a minimum data size of 12; any smaller data size results in a commit error. The minimum data size for TCP probe packets is 1.
>
> When you configure either **probe-type udp-ping** or **probe-type udp-ping-timestamp** along with the **one-way-hardware-timestamp** command, the value for the **destination-port** can be only 7. A constraint check prevents you for configuring any other value for the destination port in this case.

- To specify the routing instance used by ICMP probes, include the **routing-instance** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. The default routing instance is Internet routing table **inet.0**.
- To specify the source IP address used for ICMP probes, include the **source-address** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. If the source IP address is not one of the router's assigned addresses, the packet uses the outgoing interface's address as its source.
- Starting in Junos OS Release 16.1R1, to specify the source IPv6 address to be used for RPM probes that are sent from the RPM client (the device that originates the RPM packets) to the RPM server (the device that receives the RPM probes), include the **inet6-options source-address *ipv6-address* statement** at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. If the source IPv6 address is not one of the router's or switch's assigned addresses, the packet use the outgoing interface's address as its source.
- To specify the destination address used for the probes, include the **target** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level.
  - For HTTP probe types, specify a fully formed URL that includes **http://** in the URL address.
  - For all other probe types, specify an IP version 4 (IPv4) or IP version 6 (IPv6) (IPv6 support starts in Junos OS release 16.1R1) address for the target host.
- To specify the time to wait between tests, include the **test-interval** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. Specify a value from **0** through **86400** seconds. A value of 0 seconds causes the RPM test to stop after one iteration. The default value is 1.
- To specify thresholds used for the probes, include the **thresholds** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. A system log message is generated when the configured threshold is exceeded. Likewise, an SNMP trap (if configured) is generated when a threshold is exceeded. The following options are supported:
  - **egress-time**—Measures maximum source-to-destination time per probe.
  - **ingress-time**—Measures maximum destination-to-source time per probe.
  - **jitter-egress**—Measures maximum source-to-destination jitter per test.

- **jitter-ingress**—Measures maximum destination-to-source jitter per test.
- **jitter-rtt**—Measures maximum jitter per test, from 0 through 60000000 microseconds.
- **rtt**—Measures maximum round-trip time per probe, in microseconds.
- **std-dev-egress**—Measures maximum source-to-destination standard deviation per test.
- **std-dev-ingress**—Measures maximum destination-to-source standard deviation per test.
- **std-dev-rtt**—Measures maximum standard deviation per test, in microseconds.
- **successive-loss**—Measures successive probe loss count, indicating probe failure.
- **total-loss**—Measures total probe loss count indicating test failure, from 0 through 15.

- Traps are sent if the configured threshold is met or exceeded. To set the trap bit to generate traps, include the **traps** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level. The following options are supported:
  - **egress-jitter-exceeded**—Generates traps when the jitter in egress time threshold is met or exceeded.
  - **egress-std-dev-exceeded**—Generates traps when the egress time standard deviation threshold is met or exceeded.
  - **egress-time-exceeded**—Generates traps when the maximum egress time threshold is met or exceeded.
  - **ingress-jitter-exceeded**—Generates traps when the jitter in ingress time threshold is met or exceeded.
  - **ingress-std-dev-exceeded**—Generates traps when the ingress time standard deviation threshold is met or exceeded.
  - **ingress-time-exceeded**—Generates traps when the maximum ingress time threshold is met or exceeded.
  - **jitter-exceeded**—Generates traps when the jitter in round-trip time threshold is met or exceeded.
  - **probe-failure**—Generates traps for successive probe loss thresholds crossed.
  - **rtt-exceeded**—Generates traps when the maximum round-trip time threshold is met or exceeded.
  - **std-dev-exceeded**—Generates traps when the round-trip time standard deviation threshold is met or exceeded.
  - **test-completion**—Generates traps when a test is completed.
  - **test-failure**—Generates traps when the total probe loss threshold is met or exceeded.

Release History Table

Release	Description
18.2R1	Starting in Junos OS Release 18.2R1, you can also optimize the CLI configuration for RPM tests for IPv6.
18.1R1	Starting in Junos OS Release 18.1R1, you can configure the generation of **icmp6-ping** RPM probes on an MS-MPC or MS-MIC.
18.1R1	Starting in Junos OS Release 18.1R1, you can use additional filters to limit the output of the **show services rpm probe-results** and **show services rpm history-results** commands for RPM probes generated on an MS-MPC or MS-MIC.
17.4R1	Starting in Junos OS Release 17.4R1, you can optimize the CLI configuration for RPM tests for IPv4.
17.3R3	Starting in Junos OS Release 17.3R3, the delegate probes are distributed evenly across the interval of 3 seconds to avoid the packet bursts in the network due to real-time performance monitoring (RPM). RPM syslogs are processed with the increase in the ramp up time of RPM delegates tests to 60 seconds. With RPM syslogs processed, the chances of multiple tests starting and ending at the same time are smaller, thus a potential restriction in **event-processing**.
17.3R1	Starting in Junos OS Release 17.3R1, PIC-based and Routing Engine-based RPM is supported for IPsec tunnels and GRE tunnels if you are using MS-MPCs or MS-MICs.
17.3R1	Starting in Junos OS Release 17.3R1, you can configure the generation of IPv4 **icmp-ping** and **icmp-ping-timestamp** RPM probes on an MS-MPC or MS-MIC, which increases the number of probes generated upto 1 million per second on every service-NPU compared to the number of probes that are generated on the Packet Forwarding Engine.
16.1	Starting in Junos OS Release 16.1R1, to specify the source IPv6 address to be used for RPM probes that are sent from the RPM client (the device that originates the RPM packets) to the RPM server (the device that receives the RPM probes), include the **inet6-options source-address *ipv6-address* statement** at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level.
16.1	For all other probe types, specify an IP version 4 (IPv4) or IP version 6 (IPv6) (IPv6 support starts in Junos OS release 16.1R1) address for the target host.

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | **578**

Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers | 635

Configuring RPM Receiver Servers

The RPM TCP and UDP probes are proprietary to Juniper Networks and require a receiver to receive the probes. To configure a server to receive the probes, include the **probe-server** statement at the **[edit services rpm]** hierarchy level:

```
[edit services rpm]
probe-server {
    tcp {
        destination-interface interface-name;
        port number;
    }
    udp {
        port number;
    }
}
```

The port number specified for the UDP and TCP server can be **7** or from **49160** through **65535**.

> **NOTE:** The **destination-interface** statement is not supported on PTX Series Packet Transport routers.

When you configure either **probe-type udp-ping** or **probe-type udp-ping-timestamp** along with the **one-way-hardware-timestamp** command, the value for the **destination-port** can be only 7. A constraint check prevents you for configuring any other value for the destination port in this case.

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | 578

Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers | 635

Limiting the Number of Concurrent RPM Probes on M, MX, T and PTX Routers and EX Series Switches

To configure the maximum number of concurrent probes allowed, include the **probe-limit** statement at the **[edit services rpm]** hierarchy level:

```
probe-limit limit;
```

Specify a limit from **1** through **500**. The default maximum number is 100.

Starting in Junos OS Release 17.2R2 and 17.3R1 for MX Series routers only, the probe-limit is 1 through 2000.

Release History Table

Release	Description
17.2R2	Starting in Junos OS Release 17.2R2 and 17.3R1 for MX Series routers only, the probe-limit is 1 through 2000.

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | **578**

Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers | **635**

Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches

To account for latency in the communication of probe messages, you can enable timestamping of the probe packets. You can timestamp the following RPM probe types: **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, and **udp-ping-timestamp**.

On M Series and T Series routers with an MS-PIC, on MX Series routers with an MS-DPC, MS-MIC, or MS-MPC, and on EX Series switches, you can enable hardware timestamping of RPM probe messages. The timestamp is applied on both the RPM client router (the router or switch that originates the RPM probes) and the RPM probe server and applies only to IPv4 traffic. It is supported on the following:

- Layer 2 service package on MS-PICs, MS-DPCs, MS-MPCs, and MS-MICs.
- Layer 3 service package on MS-PICs, MS-DPCs, MS-MPCs, and MS-MICs.
- Extension-provider services package on M Series, MX Series, and T Series services PICs that support the Extension-Provider packages (In Junos OS releases earlier than Release 12.3, the extension-provider packages were variously referred to as Junos Services Framework (JSF), MP-SDK, and eJunos.)
- Layer 2, Layer 3, SDK Services, and PFE RPM timestamping interoperate with each other. Here, the RPM client can be on the Layer 3 **sp-** interface and the RPM server can be on an SDK Services package.

Two-way timestamping is available on **sp-** and **ms-** interfaces. To configure two-way timestamping on M Series and T Series routers, include the **destination-interface** statement at the **[edit services rpm probe *probe-owner* test *test-name*]** hierarchy level:

```
destination-interface sp-fpc/pic/port.logical-unit
destination-interface ms-fpc/pic/port.logical-unit
```

Specify the RPM client router and the RPM server router on the services logical interface or the multiservices interface by including the **rpm** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
rpm (client | server);
```

The logical interface must be dedicated to the RPM task. It requires configuration of the **family inet** statement and a **/32** address, as shown in the example. This configuration is also needed for other services such as NAT and stateful firewall. You cannot configure RPM service on **unit 0** because RPM requires a dedicated logical interface; the same unit cannot support both RPM and other services. Because active flow monitoring requires **unit 0**, but RPM can function on any logical interface, a constraint check prevents you from committing an RPM configuration there.

On MX Series routers, on M320 Series routers using the Enhanced Queuing MPC, and on EX Series switches, you include the **hardware-timestamp** statement at the **[edit services rpm probe *probe-name* test *test-name*]** hierarchy level to specify that the probes are to be timestamped in the Packet Forwarding Engine host processor:

On MX series routers and EX Series switches you can include the **hardware-timestamp** statement at the **[edit services rpm probe *probe-name* test *test-name*]** hierarchy level to specify that the probes are to be timestamped in the Packet Forwarding Engine host processor. On MX series routers, the feature is supported on following types of line cards:

- DPC
- DPCE
- MPC1
- MPC2
- MPC3
- MPC4
- MPC5
- MPC6
- MPC7

```
hardware-timestamp;
```

On the client side, these probes are timestamped in the Packet Forwarding Engine host processor on the egress DPC on the MX Series or M320 Series router or EX Series switch originating the RPM probes (RPM client). On the responder side (RPM server), the RPM probes to be timestamped are handled by the Packet Forwarding Engine host processor, which generates the response instead of the RPM process. The RPM probes are timestamped only on the router that originates them (RPM client). As a result, only round-trip time is measured for these probes.

When using the **hardware-timestamp**, the **data-size** value for the probe must be at least 100 bytes smaller than the default MTU of the interface of the RPM client interface (see "Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches" on page 591). If hardware timestamping of RPM probe messages is enabled, the maximum data size that you can configure by using the data-size statement is limited to 1400.

> **NOTE:** The Packet Forwarding Engine-based RPM feature does not support any stateful firewall configurations. If you need to combine RPM timestamping with a stateful firewall, use the interface-based RPM timestamping service described earlier in this section. MS-DPCs support stateful firewall processing as well as RPM timestamping.

To configure one-way timestamping, you must also include the **one-way-hardware-timestamp** statement at the **[edit services rpm probe *probe-owner* test *test-name*]** hierarchy level:

```
one-way-hardware-timestamp;
```

NOTE: If you configure RPM probes for a services interface (**sp-**), you need to announce local routes in a specific way for the following routing protocols:

- For OSPF, you can announce the local route by including the services interface in the OSPF area. To configure this setting, include the **interface sp-*fpc*/*pic*/*port*** statement at the **[edit protocols ospf area *area-number*]** hierarchy level.
- For BGP and IS-IS, you must export interface routes and create a policy that accepts the services interface local route. To export interface routes, include the **point-to-point** and **lan** statements at the **[edit routing-options interface-routes family inet export]** hierarchy level. To configure an export policy that accepts the services interface local route, include the **protocol local**, **rib inet.0**, and **route-filter *sp-interface-ip-address*/32 exact** statements at the **[edit policy-options policy-statement *policy-name* term *term-name* from]** hierarchy level and the **accept** action at the **[edit policy-options policy-statement *policy-name* term *term-name* then]** hierarchy level. For the export policy to take effect, apply the policy to BGP or IS-IS with the **export *policy-name*** statement at the **[edit protocols *protocol-name*]** hierarchy level.

For more information about these configurations, see the *Routing Policies, Firewall Filters, and Traffic Policers User Guide* or the *Junos OS Routing Protocols Library*.

Routing the probe packets through the multiservices card also enables you to filter the probe packets to particular queues. The following example shows the RPM configuration and the filter that specifies queuing:

```
services rpm {
    probe p1 {
        test t1 {
            probe-type icmp-ping;
            target address 10.8.4.1;
            probe-count 10;
            probe-interval 10;
            test-interval 10;
            dscp-code-points af11;
            data-size 100;
            destination-interface sp-1/2/0.0;
        }
    }
}
firewall {
    filter f1 {
        term t1 {
            from {
                dscp af11;
```

```
            }
            then {
                forwarding-class assured-forwarding;
            }
        }
    }
}
interfaces sp-1/2/0 {
    unit 2 {
        rpm client;
        family inet {
            address 10.8.4.2/32;
            filter {
                input f1;
            }
        }
    }
}
interfaces sp-1/2/1 {
    unit 2 {
        rpm server;
        family inet {
            address 10.8.3.2/32;
            filter {
                input f1;
            }
        }
    }
}
```

For more information about firewall filters, see the *Routing Policies, Firewall Filters, and Traffic Policers User Guide*; for more information about queuing, see the *Class of Service User Guide (Routers and EX9200 Switches)*.

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | **578**

Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers | **635**

Analyzing Network Efficiency in IPv6 Networks on MX Series Routers Using RPM Probes

Real-time performance monitoring (RPM) is a mechanism that enables you to monitor network performance in real time and to assess and analyze network efficiency. Typically, network performance is assessed in real time based on the jitter, delay, and packet loss experienced on the network. RPM is a service available in Junos OS that enables a router to measure metrics such as round-trip delays and unanswered echo requests. To compute these parameters, RPM exchanges a set of probes with other IP hosts in the network for monitoring and network tracking purposes. These probes are sent from a source node to other destination devices in the network that require tracking. Data such as transit delay and jitter can be collected from these probes, and this data can be used to provide an approximation of the delay and jitter experienced by live traffic in the network. Different live traffic metrics such as round-trip time (RTT), positive egress jitter, negative egress jitter, positive ingress jitter, negative ingress jitter, positive round-trip jitter, and negative round-trip jitter can be obtained from the results of the RPM test. RPM calculates minimum, maximum, average, peak-to-peak, standard deviation, and sum calculations for each of these measurements. RPM probes can also be used to verify the path between BGP neighbors.

Starting with Junos OS release 16.1, the RPM client router (the router or switch that originates the RPM probes) can send probe packets to the RPM probe server (the device that receives the RPM probes) that contains an IPv6 address. To specify the destination IPv6 address used for the probes, include the **target (url *ipv6-url* | address *ipv6-address*)** statement at the **[edit services rpm probe owner test *test-name*]** hierarchy level. The protocol family for IPv6 is named inet6.

```
[edit services rpm]
probe owner {
    test test-name {
        target (url ipv6-url | address ipv6-address);
    }
}
```

To specify the IPv6 protocol-related settings and the source IPv6 address of the client from which the RPM probes are sent, include the **inet6-options source-address *ipv6-address*** statement at the **[edit services rpm probe owner test *test-name*]** hierarchy level. A probe request is a standard packet with corresponding TCP, UDP, and ICMP headers over the IPv6 header. No RPM header is appended to the standard packet for Routing Engine-based RPM implementation. A probe response is also a standard packet with corresponding TCP, UDP, and ICMP headers over the IPv6 header. No RPM header is appended to the standard packet for Routing Engine-based RPM implementation.

```
[edit services rpm]
probe owner {
    test test-name {
        inet6-options source-address ipv6-address;
```

```
    }
}
```

The output of the **show services rpm probe-results owner *probe-name* test *test-name*** and **show services rpm history-results owner *owner* test *name*** commands that display the results of the most recent RPM probes and results of historical RPM probes respectively have been enhanced to display the target address as IPv6 address and other IPv6 information for probes sent to IPv6 servers or destinations. The existing SNMP Get requests and traps for IPv6 are applicable for IPv6 probes. The target type field in the SNMP set operation contains IPv6 source and destination addresses.

Guidelines for Configuring RPM Probes for IPv6 Destinations

Keep the following points in mind when you configure IPv6 addresses for RPM destinations or servers:

- Only Routing Engine-based RPM is supported for IPv6 targets including VRF support, specification of the size of the data portion of ICMP probes, data pattern, and traffic class.
- You can configure probes with a combination of IPv4 and IPv6 tests. However, a test can be either IPv4 or IPv6-based at a point in time. The OS impacts the accuracy of the measurements because the variability factor introduced by the general OS that performs the system processing proved is significantly larger than the amount of time spent by the packet traversing on the wire. This condition causes round-trip time (RTT) spikes to be seen even with a single test.
- Routing Engine-based RPM does not support one-way hardware-based timestamping.
- One-way measurements are not supported here because timestamping is done only on the RPM client side.
- The maximum number of concurrent probes allowed (by including the **probe-limit** statement at the **[edit services rpm]** hierarchy level) is 1000. We recommend that the limit on concurrent probes be set as 10. Higher concurrent probes can result in higher spikes. The maximum number of tests you can configure is 1000. RPM cannot be configured on logical systems. SNMP set operation is permitted only on ICMP probes and it is not supported for other type of probes.
- The **hardware-timestamp** and **one-way-hardware-timestamp** statements at the **[edit services rpm probe owner test *test-name*]** hierarchy level are not supported for IPv6.
- You cannot specify the **icmp-ping** (which sends ICMP echo requests to a target address) and the **icmp-ping-timestamp** (which sends ICMP timestamp requests to a target address) options with the probe-type statement at the **[edit services rpm probe owner test *test-name*]** hierarchy level.
- Some of the RPM problems can resolved by restarting the SNMP remote operations process (rmopd) on the Routing Engine by using the restart remote-operations command. If RPM needs to be disabled, the rpm statement at the [edit services] hierarchy level needs to be deleted or deactivated. PIC, Packet Forwarding Engine, and lookup chip (LU) based RPM implementation for IPv6 are not supported.

- The following table describes the IPv6 special address prefixes that are not supported.

IPV6 Address Type	IPV6 Address Prefix
Node-Scoped Unicast	::1/128 is the loopback address ::/128 is the unspecified address
IPv4-Mapped Addresses	::FFFF:0:0/96
IPv4-Compatible Addresses	:<ipv4-address>/96
Link-Scoped Unicast	fe80::/10
Unique-Local	fc00::/7
Documentation Prefix	2001:db8::/32
6to4	2002::/16
6bone	5f00::/8
ORCHID	2001:10::/28
Teredo	2001::/32
Default Route	::/0
Multicast	ff00::/8

- The current scaling number for IPv4 probes is a maximum of 500 concurrent probes and the limit on the maximum number of configurable tests is 1000. These scaling parameters are applicable for IPv6 probes. The same scaling limits are applicable, even in cases where both IPv4-based tests and IPv6-based tests are run at the same time.
- The minimum rate of probes is 1 probe per second and the maximum interval between tests is 86400 seconds. These scaling and performance numbers vary based on whether the Two-Way Active Measurement Protocol (TWAMP) server and client are configured on the same router. This condition occurs because the TWAMP server/client has packet processing in RMOPD and it competes with RPM functionality in the same process. The RTT of IPv6-based RPM and ping utilities must be equivalent for data size. In Routing Engine-based RPM implementation, RTT spikes are seen owing to various queuing delays introduced in the system. This behavior can be noticed even with a single test.

- Some of the TCP and UDP ports might be opened to communicate between the RPM server and RPM client. Therefore, we recommend that you use firewalls and distributed denial-of-service (DDoS) attack filters to ensure that no security threats are possible by some third-party attackers or hackers.
- The different packet types that can be used within the probe include:
  - ICMP6 echo
  - UDP echo
  - UDP timestamp

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches

IN THIS SECTION

Two-Way Active Measurement Protocol (TWAMP) support and configuration varies for hardware platform, physical interfaces, or virtual physical (services) interfaces. To enable some features, you configure TWAMP under the real-time performance monitoring (RPM) stanza. However, support for RPM is not always an indicator of TWAMP support on a particular combination of platform and line card. The time stamps used in RPM and TWAMP added in different places, depending on the hardware configuration.

For example, different hardware components perform timestamping either in the Routing Engine, the Packet Forwarding Engine, or the line card such as a Multservices Physical Interface Card (MS-PIC), Multservices Modular Interface Card (MS-MIC), Multservices Modular PIC Concentrator (MS-MPC), or Multservices Dense Port Concentrator (MS-DPC).

NOTE: TWAMP is not supported on PTX Series Packet Transport routers.

Table 100 on page 611 shows the relationship between RPM client and server support, TWAMP client (with the control component) and TWAMP server (with the responder component) support, and the hardware that performs timestamping.

Table 100: TWAMP Feature Support and Hardware

TWAMP Feature Support	Routing Engine Timestamp	MS-PIC/MS-DPC Timestamp	MS-MIC/MS-MPC Timestamp	Packet Forwarding Engine (ukernel) Timestamp	Packet Forwarding Engine (LU) Timestamp (**si-** interface)
RPM Client	Yes	Yes	Yes	Yes	No
RPM Server	Yes	Yes	Yes	Yes	No
TWAMP Client	No	No	No	No	Yes
TWAMP Server	No	Yes	No	Yes (No responder configuration needed)	Yes

NOTE: Support for the services interfaces (**sp-**, **ms-**, and **si-** interfaces) are all slightly different.

For more information on TWAMP, see RFC 5357, *A Two-Way Active Measurement Protocol (TWAMP)*.

The TWAMP configuration process includes the following tasks:

Configuring a TWAMP Server

With the exception of physical interfaces, TWAMP server configuration requires the following minimum configuration at the **[edit services rpm twamp]** hierarchy level:

```
user@router# show services rpm twamp
server {
    authentication-mode mode;
    client-list list-name {
        address ip-address;
    }
    port 662;
}
```

You cannot use the following addresses for the client-list source IP address used for probes:

- 0.0.0.0
- 127.0.0.0/8 (loopback)
- 224.0.0.0/4 (multicast)
- 255.255.255.255. (broadcast)

You can configure more than one client, and you can change the TWAMP listening port as long as the change is coordinated with the TWAMP client.

For **si-** or **sp-** services interfaces, TWAMP server configuration requires the following statements at the **[edit interfaces *service-interface-name*]** hierarchy level:

```
user@router# show interfaces si-0/0/0
unit 0{
    rpm twamp-server;
    family inet {
        address 10.10.10.1/24;
    }
}
```

```
user@router# show interfaces sp-0/0/0
unit 0{
    rpm twamp-server;
    family inet {
        address 10.20.20.1/24;
    }
}
```

To configure a TWAMP server on an inline services (**si-**) interface, configure the amount of bandwidth reserved on each Packet Forwarding Engine for tunnel traffic using inline services by including the **bandwidth (1g | 10g)** statement at the **[edit chassis fpc slot-number pic number inline-services]** hierarchy level. Specify the service PIC (**sp-**) logical interface that provides the TWAMP service by including the **twamp-server** statement at the **[edit interfaces sp-*fpc/pic/port* unit *logical-unit-number* family inet]** hierarchy level.

The twamp-server statement is not required for physical interface TWAMP server configuration.

Many other TWAMP server parameters are optional. See the TWAMP **server** configuration statements for details.

SEE ALSO

Configuring a TWAMP Client

The **si-** interfaces are virtual physical interfaces that respond as a TWAMP server. However, you can also configure services interfaces to act as the TWAMP client, which performs TWAMP controller role.

To configure a services interface as a TWAMP client, you configure the service parameters and the service interface as a TWAMP client.

To configure the TWAMP client service, include the **client** statement and related parameters at the **[edit services rpm twamp]** hierarchy level.

```
user@router# show services rpm twamp
client {
   [. . .]
}
```

There are many parameters possible for TWAMP client configuration. See the configuration statement topics and examples for details.

To configure the TWAMP client services interface, include the **rpm twamp-client** statement at the **[edit interfaces *si-interface-name*]** hierarchy level:

```
user@router# show interfaces si-0/0/0
unit 0 {
   family inet;
}
unit 10 {
   rpm twamp-client;
   family inet {
      address 10.30.30.1/24
   }
}
```

NOTE: You cannot configure the TWAMP client on unit 0 of a service interface. If you try, you will receive a configuration error.

RELATED DOCUMENTATION

- Understanding Two-Way Active Measurement Protocol on Routers | **582**
- Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**
- Configuring RPM Receiver Servers | **601**
- Limiting the Number of Concurrent RPM Probes on M, MX, T and PTX Routers and EX Series Switches | **602**
- Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches | **602**
- Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers | **635**

Configuring TWAMP Client and TWAMP Server to Reconnect Automatically After TWAMP Server Unavailability

You can run TWAMP client automatically without any manual intervention during network failures or configuration changes. In case of a network outage or connection loss between a TWAMP client and TWAMP server, all the affected TWAMP TCP control connections and UDP test-sessions are lost. At each test-interval, the TWAMP client continues to send the control packets to re-establish connectivity with TWAMP server till is successful. All the statistics will be maintained during that network failure.

To configure the TWAMP client:

1. Configure the interfaces.

```
[edit interfaces]
user@router1# set si-2/2/0 unit 0 family inet
user@router1# set si-2/2/0 unit 10 rpm twamp-client
user@router1# set si-2/2/0 unit 10 family inet address 192.168.20.1/32
```

2. Configure the chassis.

```
[edit chassis]
```

```
user@router1# set fpc 2 pic 2 inline-services bandwidth 1g
```

3. Configure the services.

```
[edit services]
user@router1# set rpm twamp client control-connection c1 destination-interface si-2/2/0.10
user@router1# set rpm twamp client control-connection c1 persistent-results
user@router1# set rpm twamp client control-connection c1 history-size 500
user@router1# set rpm twamp client control-connection c1 routing instance IN
user@router1# set rpm twamp client control-connection c1 target-address 192.0.2.2
user@router1# set rpm twamp client control-connection c1 tcp-keepidle 20
user@router1# set rpm twamp client control-connection c1 tcp-keepintvl 4
user@router1# set rpm twamp client control-connection c1 tcp-keepcnt 10
user@router1# set rpm twamp client control-connection c1 test-interval 4
user@router1# set rpm twamp client control-connection c1 traps control-connection-closed
user@router1# set rpm twamp client control-connection c1 test-session t1 target-address 192.0.2.2
user@router1# set rpm twamp client control-connection c1 test-session t1 data-fill-with-zeros
user@router1# set rpm twamp client control-connection c1 test-session t1 data-size 1400
user@router1# set rpm twamp client control-connection c1 test-session t1 probe-count 20
user@router1# set rpm twamp client control-connection c1 test-session t1 probe-interval 1
user@router1# set rpm twamp client control-connection c1 test-session t2 target-address 192.0.2.2
user@router1# set rpm twamp client control-connection c1 test-session t1 data-fill-with-zeros
user@router1# set rpm twamp client control-connection c1 test-session t1 data-size 1400
user@router1# set rpm twamp client control-connection c1 test-session t1 probe-count 20
user@router1# set rpm twamp client control-connection c1 test-session t1 probe-interval 1
user@router1# set rpm twamp client control-connection c1 test-session t1 thresholds total-loss 10
user@router1# set rpm twamp client control-connection c1 test-session t1 thresholds jitter-gress 20
user@router1# set rpm twamp client control-connection c1 test-session t2 target-address 192.0.3.1
user@router1# set rpm twamp client control-connection c1 test-session t2 data-fill-with-zeros
user@router1# set rpm twamp client control-connection c1 test-session t2 data-size 1400
user@router1# set rpm twamp client control-connection c1 test-session t2 probe-count 15
user@router1# set rpm twamp client control-connection c1 test-session t2 probe-interval 1
user@router1# set rpm twamp client control-connection c1 test-session t2 thresholds total-loss 10
user@router1# set rpm twamp client control-connection c1 test-session t2 thresholds jitter-gress 20
```

To configure the TWAMP server:

1. Configure the interfaces.

```
[edit interfaces]
user@router1#set si-1/1/0 unit 30 family inet
user@router1#set si-1/1/0 unit 30 rpm twamp-server
```

```
user@router1#set si-1/1/0 unit 30 family inet address 192.02.2/24
```

2. Configure the chassis.

```
[edit chassis]
user@router1# set fpc 1pic 1 inline-services bandwidth 1g
```

3. Configure the services.

```
[edit services]
user@router1# set rpm twamp server tcp-keepidle 200
user@router1# set rpm twamp server tcp-keepintvl 20
user@router1# set rpm twamp server tcp-keepcnt 210
user@router1# set rpm twamp server authentication-mode none
user@router1# set rpm twamp server server-inactivity-timeout 5
user@router1# set rpm twamp server reflector-inactivity-timeout 15
user@router1# set rpm twamp server max-connection-duration 0
user@router1# set rpm twamp server maximum-sessions 100
user@router1# set rpm twamp server maximum-sessions-per-connection 50
user@router1# set rpm twamp server maximum-connections 500
user@router1# set rpm twamp server maximum-connections-per-client 500
user@router1# set rpm twamp server port 862
user@router1# set rpm twamp server client-list Client1 address 192.168.20.1/24
```

When the TWAMP server is reachable, the output is as follows. The **TWAMP-Server-Status** is **Connected** and the **Number-Of-Retries-With-TWAMP-Server** is 1

user@router1# run show services rpm twamp client probe-results |no-more

```
Jan 11 11:43:42
    Owner: c1, Test: t1
    server-address: 192.0.2.2, server-port: 862, Client address: 192.168.20.1,
Client port: 58991
    TWAMP-Server-Status: Connected, Number-Of-Retries-With-TWAMP-Server: 1
    Routing Instance Name: IN
    Destination interface name: si-2/2/0.10
    Test size: 20 probes
    Probe results:
      Response received
      Probe sent time: Fri Jan 11 11:43:41 2019
      Probe rcvd/timeout time: Fri Jan 11 11:43:41 2019
```

```
      Rtt: 57 usec, Egress jitter: 1 usec, Ingress jitter: -1 usec, Round trip
jitter: 0 usec
      Egress interarrival jitter: 43 usec, Ingress interarrival jitter: 43 usec,
Round trip interarrival jitter: 1 usec
    Results over current test:

.......
........
    Owner: c1, Test: t2
    server-address: 192.0.2.2, server-port: 862, Client address: 192.168.20.1,
Client port: 58991
    TWAMP-Server-Status: Connected, Number-Of-Retries-With-TWAMP-Server: 1
    Routing Instance Name: IN
    Destination interface name: si-2/2/0.10
    Test size: 15 probes
   Probe results:
      Response received
      Probe sent time: Fri Jan 11 11:43:36 2019
      Probe rcvd/timeout time: Fri Jan 11 11:43:36 2019
      Rtt: 58 usec, Egress jitter: 1 usec, Ingress jitter: -1 usec, Round trip
jitter: 0 usec
      Egress interarrival jitter: 28 usec, Ingress interarrival jitter: 28 usec,
Round trip interarrival jitter: 0 usec
    Results over current test:
      Probes sent: 15, Probes received: 15, Loss percentage: 0.000000
      Measurement: Round trip time
        Samples: 15, Minimum: 57 usec, Maximum: 59 usec, Average: 58 usec, Peak
to peak: 2 usec, Stddev: 1 usec, Sum: 866 usec
      Measurement: Positive egress jitter

........
    Measurement: Round trip time
        Samples: 105, Minimum: 57 usec, Maximum: 59 usec, Average: 58 usec, Peak
to peak: 2 usec, Stddev: 1 usec, Sum: 6062 usec
      Measurement: Positive egress jitter
        Samples: 77, Minimum: 0 usec, Maximum: 398 usec, Average: 12 usec, Peak
to peak: 398 usec, Stddev: 63 usec, Sum: 925 usec
      Measurement: Negative egress jitter
        Samples: 18, Minimum: 16 usec, Maximum: 431 usec, Average: 69 usec, Peak
to peak: 415 usec, Stddev: 91 usec, Sum: 1248 usec
      Measurement: Positive ingress jitter
        Samples: 19, Minimum: 0 usec, Maximum: 431 usec, Average: 66 usec, Peak
to peak: 431 usec, Stddev: 90 usec, Sum: 1249 usec
      Measurement: Negative ingress jitter
```

```
        Samples: 76, Minimum: 1 usec, Maximum: 397 usec, Average: 12 usec, Peak
to peak: 396 usec, Stddev: 63 usec, Sum: 922 usec
      Measurement: Positive round trip jitter
        Samples: 79, Minimum: 0 usec, Maximum: 1 usec, Average: 0 usec, Peak to
peak: 1 usec, Stddev: 0 usec, Sum: 26 usec
      Measurement: Negative round trip jitter
        Samples: 25, Minimum: 1 usec, Maximum: 1 usec, Average: 1 usec, Peak to
peak: 0 usec, Stddev: 0 usec, Sum: 25 usec
```

After the server is deactivated using the command **deactivate interfaces si-1/1/0 unit 30** , the output is as follows. The **TWAMP-Server-Status** is **Not Connected** and the **Number-Of-Retries-With-TWAMP-Server** is 12:

user@router1# run show services rpm twamp client probe-results control-connection c1 |no-more

```
Jan 11 11:48:24
    Owner: c1, Test: t1
    server-address: 192.0.2.2, server-port: 862, Client address: 192.168.20.1,
Client port: 58991
    TWAMP-Server-Status: Not Connected, Number-Of-Retries-With-TWAMP-Server: 12
    Reflector address: 192.0.2.2, Reflector port: 14779, Sender address:
192.168.20.1, sender-port: 14779
    Routing Instance Name: IN
    Destination interface name: si-2/2/0.10
    Test size: 20 probes
    Probe results:
      Response received
      Probe sent time: Fri Jan 11 11:45:38 2019
      Probe rcvd/timeout time: Fri Jan 11 11:45:38 2019
      Rtt: 55 usec, Egress jitter: -17 usec, Ingress jitter: 18 usec, Round trip
jitter: 1 usec
      Egress interarrival jitter: 37 usec, Ingress interarrival jitter: 37 usec,
Round trip interarrival jitter: 0 usec
    Results over current test:
      Probes sent: 10, Probes received: 10, Loss percentage: 0.000000
      Measurement: Round trip time

.........
   Samples: 17, Minimum: 0 usec, Maximum: 3 usec, Average: 0 usec, Peak to peak:
3 usec, Stddev: 1 usec, Sum: 4 usec
      Measurement: Negative round trip jitter
        Samples: 3, Minimum: 1 usec, Maximum: 3 usec, Average: 2 usec, Peak to
peak: 2 usec, Stddev: 1 usec, Sum: 5 usec
```

```
    Results over all tests:
      Probes sent: 210, Probes received: 210, Loss percentage: 0.000000


........

TWAMP-Server-Status: Not Connected, Number-Of-Retries-With-TWAMP-Server: 12
    Reflector address: 192.0.2.2, Reflector port: 14778, Sender address:
192.168.20.1, sender-port: 14778
    Routing Instance Name: IN
    Destination interface name: si-2/2/0.10
    Test size: 15 probes
    Probe results:
      Response received
      Probe sent time: Fri Jan 11 11:45:38 2019
      Probe rcvd/timeout time: Fri Jan 11 11:45:38 2019
      Rtt: 58 usec, Egress jitter: -18 usec, Ingress jitter: 19 usec, Round trip
jitter: 0 usec

..........
    Results over all tests:
      Probes sent: 160, Probes received: 160, Loss percentage: 0.000000
      Measurement: Round trip time
        Samples: 160, Minimum: 57 usec, Maximum: 59 usec, Average: 58 usec, Peak
to peak: 2 usec, Stddev: 1 usec, Sum: 9232 usec
      Measurement: Positive egress jitter
        Samples: 119, Minimum: 0 usec, Maximum: 398 usec, Average: 12 usec, Peak
to peak: 398 usec, Stddev: 62 usec, Sum: 1398 usec
      Measurement: Negative egress jitter
        Samples: 27, Minimum: 16 usec, Maximum: 431 usec, Average: 64 usec, Peak
to peak: 415 usec, Stddev: 76 usec, Sum: 1723 usec
      Measurement: Positive ingress jitter
        Samples: 28, Minimum: 0 usec, Maximum: 431 usec, Average: 62 usec, Peak
to peak: 431 usec, Stddev: 76 usec, Sum: 1727 usec
      Measurement: Negative ingress jitter
        Samples: 118, Minimum: 1 usec, Maximum: 397 usec, Average: 12 usec, Peak
to peak: 396 usec, Stddev: 62 usec, Sum: 1400 usec
      Measurement: Positive round trip jitter
        Samples: 120, Minimum: 0 usec, Maximum: 1 usec, Average: 0 usec, Peak to
peak: 1 usec, Stddev: 0 usec, Sum: 39 usec
      Measurement: Negative round trip jitter
        Samples: 39, Minimum: 1 usec, Maximum: 1 usec, Average: 1 usec, Peak to
peak: 0 usec, Stddev: 0 usec, Sum: 39 usec
```

After activating the server using the **activate interfaces si-1/1/0 unit 30** command the output is as follows. The **TWAMP-Server-Status** is **Connected** and the **Number-Of-Retries-With-TWAMP-Server** is 12.

user@router1# run show services rpm twamp client probe-results control-connection c1 |no-more

```
Jan 11 11:48:50
    Owner: c1, Test: t1
    server-address: 192.0.2.2, server-port: 862, Client address: 192.168.20.1,
Client port: 58991
    TWAMP-Server-Status: Connected, Number-Of-Retries-With-TWAMP-Server: 12
    Reflector address: 192.0.2.2, Reflector port: 14963, Sender address:
192.168.20.1, sender-port: 14963
    Routing Instance Name: IN
    Destination interface name: si-2/2/0.10
    Test size: 20 probes
    Probe results:
      Response received
      Probe sent time: Fri Jan 11 11:48:50 2019
      Probe rcvd/timeout time: Fri Jan 11 11:48:50 2019

.....
Results over all tests:
      Probes sent: 218, Probes received: 218, Loss percentage: 0.000000
      Measurement: Round trip time
        Samples: 218, Minimum: 54 usec, Maximum: 59 usec, Average: 56 usec, Peak
to peak: 5 usec, Stddev: 1 usec, Sum: 12160 usec

......
    Owner: c1, Test: t2
    server-address: 192.0.2.2, server-port: 862, Client address: 192.168.20.1,
Client port: 58991
    TWAMP-Server-Status: Connected, Number-Of-Retries-With-TWAMP-Server: 12
    Reflector address: 192.0.2.2, Reflector port: 14962, Sender address:
192.168.20.1, sender-port: 14962
    Routing Instance Name: IN
    Destination interface name: si-2/2/0.10
    Test size: 15 probes
    Probe results:
      Response received
      Probe sent time: Fri Jan 11 11:48:50 2019
      Probe rcvd/timeout time: Fri Jan 11 11:48:50 2019
      Rtt: 57 usec, Egress jitter: 2 usec, Ingress jitter: -3 usec,
......
Results over all tests:
      Probes sent: 168, Probes received: 168, Loss percentage: 0.000000
```

```
      Measurement: Round trip time
        Samples: 168, Minimum: 57 usec, Maximum: 59 usec, Average: 58 usec, Peak
to peak: 2 usec, Stddev: 1 usec, Sum: 9691 usec
      Measurement: Positive egress jitter
        Samples: 124, Minimum: 0 usec, Maximum: 398 usec, Average: 11 usec, Peak
to peak: 398 usec, Stddev: 61 usec, Sum: 1406 usec
      Measurement: Negative egress jitter
        Samples: 29, Minimum: 16 usec, Maximum: 431 usec, Average: 62 usec, Peak
to peak: 415 usec, Stddev: 74 usec, Sum: 1806 usec
      Measurement: Positive ingress jitter
        Samples: 30, Minimum: 0 usec, Maximum: 431 usec, Average: 60 usec, Peak
to peak: 431 usec, Stddev: 74 usec, Sum: 1811 usec
      Measurement: Negative ingress jitter
        Samples: 123, Minimum: 1 usec, Maximum: 397 usec, Average: 11 usec, Peak
to peak: 396 usec, Stddev: 61 usec, Sum: 1410 usec
      Measurement: Positive round trip jitter
        Samples: 125, Minimum: 0 usec, Maximum: 1 usec, Average: 0 usec, Peak to
peak: 1 usec, Stddev: 0 usec, Sum: 42 usec
      Measurement: Negative round trip jitter
        Samples: 42, Minimum: 1 usec, Maximum: 1 usec, Average: 1 usec, Peak to
peak: 0 usec, Stddev: 0 usec, Sum: 42 usec
```

RELATED DOCUMENTATION

- tcp-keepcnt | **1157**
- tcp-keepintvl | **1159**
- tcp-keepidle | **1158**
- persistent-results | **1079**
- Understanding TWAMP Auto-Restart | **587**

Example: Configuring TWAMP Client and Server on MX Series Routers

IN THIS SECTION

This example shows how to configure TWAMP client and server and contains the following sections.

Requirements

This example uses the following hardware and software components:

- MX Series routers.
- Junos OS Release 15.1 or later.

Overview

This example explains the Two-Way Active Measurement Protocol (TWAMP). TWAMP is an open protocol for measuring network performance between any two devices supporting the TWAMP protocol. The TWAMP-Control protocol is used to set up performance measurement sessions. The TWAMP-Test protocol is used to send and receive performance measurement probes.

The TWAMP architecture is composed of the following entities that are responsible for starting a monitoring session and exchanging packets:

- The control client initiates all requested test sessions with a start sessions message, and the server acknowledges. When necessary, the control client sends a message to stop all test sessions.
- The session sender and the session reflector exchange test packets according to the TWAMP-Test protocol for each active session. On receiving a TWAMP-Test packet, the session reflector only reflects a measurement packet and does not collect packet statistics in TWAMP.

The TWAMP server is an end system that manages one or more TWAMP sessions and is also capable of configuring per-session ports. The server listens on the TCP port. The session reflector and server make up the TWAMP responder in an IP service-level agreement operation.

For 15.1, both the control client and session sender would be residing on the same Juniper router. The client design does not mandate the server and the session reflector to be on the same system. Hence the Juniper TWAMP client will also be capable of working with a third-party server implementation.

Configuration for TWAMP client

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, of SG1 router.

Configuring Interfaces

```
set interfaces si-4/1/0 unit 0 family inet
set interfaces si-4/1/0 unit 10 rpm twamp-client
set interfaces si-4/1/0 unit 10 family inet address 60.60.60.1/32
```

Configuring Chassis

```
set chassis fpc 4 pic 1 inline-services bandwidth 1g
```

Configuring Services

```
set services rpm twamp client control-connection c1 destination-interface si-4/1/0.10
set services rpm twamp client control-connection c1 history-size 500
set services rpm twamp client control-connection c1 target-address 70.70.70.1
set services rpm twamp client control-connection c1 test-count 1
set services rpm twamp client control-connection c1 test-interval 1
set services rpm twamp client control-connection c1 traps test-iteration-done
set services rpm twamp client control-connection c1 traps control-connection-closed
set services rpm twamp client control-connection c1 test-session t1 target-address 70.70.70.1
set services rpm twamp client control-connection c1 test-session t1 data-fill-with-zeros
set services rpm twamp client control-connection c1 test-session t1 data-size 1400
set services rpm twamp client control-connection c1 test-session t1 probe-count 55
set services rpm twamp client control-connection c1 test-session t1 probe-interval 1
```

Configuring TWAMP client

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy.

1. Configure the interfaces.

```
[edit interfaces]
user@router1# set si-4/1/0 unit 0 family inet
user@router1# set si-4/1/0 unit 10 rpm twamp-client
user@router1# set si-4/1/0 unit 10 family inet address 60.60.60.1/32
```

2. Configure the chassis.

```
[edit chassis]
user@router1# set fpc 4 pic 1 inline-services bandwidth 1g
```

3. Configure the services.

```
[edit services]
user@router1# set rpm twamp client control-connection c1 destination-interface si-4/1/0.10
user@router1# set rpm twamp client control-connection c1 history-size 500
user@router1# set rpm twamp client control-connection c1 target-address 70.70.70.1
user@router1# set rpm twamp client control-connection c1 test-count 1
user@router1# set rpm twamp client control-connection c1 test-interval 1
user@router1# set rpm twamp client control-connection c1 traps test-iteration-done
user@router1# set rpm twamp client control-connection c1 traps control-connection-closed
user@router1# set rpm twamp client control-connection c1 test-session t1 target-address 70.70.70.1
user@router1# set rpm twamp client control-connection c1 test-session t1 data-fill-with-zeros
user@router1# set rpm twamp client control-connection c1 test-session t1 data-size 1400
user@router1# set rpm twamp client control-connection c1 test-session t1 probe-count 55
user@router1# set rpm twamp client control-connection c1 test-session t1 probe-interval 1
```

Results

From the configuration mode of Router 1, confirm your configuration by entering the **show interfaces**, **show access**, and **show services** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@router1# show interface
   si-4/1/0 {
      unit 0 {
         family inet;
      }
      unit 10 {
         rpm twamp-client;
         family inet {
            address 60.60.60.1/32;
```

```
        }
      }
    }
```

```
user@router1# show services rpm twamp
client {
    control-connection c1 {
        destination-interface si-4/1/0.10;
        history-size 500;
        target-address 70.70.70.1;
        test-count 1;
        test-interval 1;
        traps {
            test-iteration-done;
            control-connection-closed;
        }
        test-session t1 {
            target-address 70.70.70.1;
            data-fill-with-zeros;
            data-size 1400;
            probe-count 55;
            probe-interval 1;
        }
    }
}
```

```
user@router1# show chassis
fpc 4 {
    pic 1 {
        inline-services {
            bandwidth 1g;
        }
    }
}
```

Configuration for TWAMP server

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, of SG1 router.

Configuring Interfaces

```
set interfaces si-2/1/0 unit 0 family inet
set interfaces si-2/1/0 unit 10 rpm twamp-server
set interfaces si-2/1/0 unit 10 family inet address 70.70.70.1/32
```

Configuring Chassis

```
set chassis fpc 2 pic 1 inline-services bandwidth 1g
```

Configuring Services

```
set services rpm twamp server authentication-mode none
set services rpm twamp server port 862
set services rpm twamp server client-list Client1 address 60.60.60.1/32
```

Configuring TWAMP server

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy.

1. Configure the interfaces.

```
[edit interfaces]
user@router1#set si-2/1/0 unit 0 family inet
user@router1#set si-2/1/0 unit 10 rpm twamp-server
user@router1#set si-2/1/0 unit 10 family inet address 70.70.70.1/32
```

2. Configure the chassis.

```
[edit chassis]
user@router1# set fpc 2 pic 1 inline-services bandwidth 1g
```

3. Configure the services.

```
[edit services]
user@router1# set rpm twamp server authentication-mode none
user@router1# set rpm twamp server port 862
user@router1# set rpm twamp server client-list Client1 address 60.60.60.1/32
```

Results

From the configuration mode of Router 1, confirm your configuration by entering the **show interfaces**, **show access**, and **show services** commands. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@router1# show interfaces
si-2/1/0 {
    unit 0 {
        family inet;
    }
    unit 10 {
        rpm twamp-server;
        family inet {
            address 70.70.70.1/32;
        }
    }
}
```

```
user@router1# show services rpm twamp server
authentication-mode none;
    port 862;
    client-list Client1 {
    address {
        60.60.60.1/32;
    }
}
```

```
user@router1# show chassis
fpc 2 {
    pic 1 {
        inline-services {
            bandwidth 1g;
        }
    }
}
```

Verification

Verifying TWAMP server sessions

Purpose

Verify that the TWAMP server sessions are established.

Action

From operational mode, enter the **show services rpm twamp server session**.

```
user@router1>show services rpm twamp server session
```

```
    Session  Connection  Sender           Sender  Reflector        Reflector
    ID       ID          address          port    address          port
          4          44  1.1.1.1           12345  192.168.219.203       890
         78          44  3.22.1.55           345  22.2.2.2            89022
        234         423  192.168.219.203    2345  2.2.22.2             3333
          5         423  221.4.1.1         82345  2.2.2.2             45909
          1         423  192.168.1.1         645  32.2.2.23            2394
```

Verifying TWAMP client sessions

Purpose

Verify that the TWAMP client sessions are established.

Action

From operational mode, enter the **show services rpm twamp client session**.

```
user@router1>show services rpm twamp client session
```

```
Connection  Session   Sender        Sender    Reflector    Reflector
      Name     Name   address         port      address         port
        c2       t1   13.13.13.13    10008   13.13.13.14       10008
```

RELATED DOCUMENTATION

request services rpm twamp start

request services rpm twamp stop

request services rpm twamp | 1247

Configuring BGP Neighbor Discovery Through RPM

BGP neighbors can be configured at the following hierarchy levels:

- **[edit protocols bgp group *group-name*]**—Default logical system and default routing instance.
- **[edit routing-instances *instance-name* protocols bgp group *group-name*]**—Default logical system with a specified routing instance.
- **[edit logical-systems *logical-system-name* protocols bgp group *group-name*]**—Configured logical system and default routing instance.
- **[edit logical-systems *logical-system-name* routing-instances *instance-name* protocols bgp group *group-name*]**—Configured logical system with a specified routing instance.

When you configure BGP neighbor discovery through RPM, if you do not specify a logical system, the RPM probe applies to configured BGP neighbors for all logical systems. If you do not specify a routing instance, the RPM probe applies to configured BGP neighbors in all routing instances. You can explicitly configure RPM probes to apply only to the default logical system, the default routing instance, or to a particular logical system or routing instance.

To configure BGP neighbor discovery through RPM, configure the probe properties at the **[edit services rpm bgp]** hierarchy:

```
data-fill data;
data-size size;
destination-port port;
history-size size;
logical-system logical-system-name [routing-instances routing-instance-name];
moving-average-size number;
probe-count count;
probe-interval seconds;
probe-type type;
routing-instances instance-name;
test-interval interval;
```

- To specify the contents of the data portion of Internet Control Message Protocol (ICMP) probes, include the **data-fill** statement at the **[edit services rpm bgp]** hierarchy level. The value can be a hexadecimal value.
- To specify the size of the data portion of ICMP probes, include the **data-size** statement at the **[edit services rpm bgp]** hierarchy level. The size can be from **0** through **65400** and the default size is **0**.
- To specify the User Datagram Protocol (UDP) port or Transmission Control Protocol (TCP) port to which the probe is sent, include the **destination-port** statement at the **[edit services rpm bgp]** hierarchy level. The **destination-port** statement is used only for the UDP and TCP probe types. The value can be **7** or from **49160** through **65535**.

- To specify the number of stored history entries, include the **history-size** statement at the **[edit services rpm bgp]** hierarchy level. Specify a value from **0** to **512**. The default is **50**.
- To specify the logical system used by ICMP probes, include the **logical-system *logical-system-name*** statement at the **[edit services rpm bgp]** hierarchy level. If you do not specify a logical system, the RPM probe applies to configured BGP neighbors for all logical systems. To apply the probe to only the default logical system, you must set the value of ***logical-system-name*** to **null**.
- To specify a number of samples for making statistical calculations, include the moving-average-size statement at the **[edit services rpm bgp]** hierarchy level. Specify a value from 0 through 255.
- To specify the number of probes within a test, include the **probe-count** statement at the **[edit services rpm bgp]** hierarchy level. Specify a value from **1** through **15**.
- To specify the time to wait between sending packets, include the **probe-interval** statement at the **[edit services rpm bgp]** hierarchy level. Specify a value from **1** through **255** seconds.
- To specify the packet and protocol contents of the probe, include the **probe-type** statement at the **[edit services rpm bgp]** hierarchy level. The following probe types are supported:
  - **icmp-ping**—Sends ICMP echo requests to a target address.
  - **icmp-ping-timestamp**—Sends ICMP timestamp requests to a target address.
  - **tcp-ping**—Sends TCP packets to a target.
  - **udp-ping**—Sends UDP packets to a target.
  - **udp-ping-timestamp**—Sends UDP timestamp requests to a target address.

    NOTE: Some probe types require additional parameters to be configured. For example, when you specify the **tcp-ping** or **udp-ping** option, you must configure the destination port using the **destination-port *port*** statement. The **udp-ping-timestamp** option requires a minimum data size of 12; any smaller data size results in a commit error. The minimum data size for TCP probe packets is 1.

- To specify the routing instance used by ICMP probes, include the **routing-instances** statement at the **[edit services rpm bgp]** hierarchy level. The default routing instance is Internet routing table **inet.0**. If you do not specify a routing instance, the RPM probe applies to configured BGP neighbors in all routing instances. To apply the RPM probe to only the default routing instance, you must explicitly set the value of ***instance-name*** to **default**.
- To specify the time to wait between tests, include the **test-interval** statement at the **[edit services bgp probe]** hierarchy level. Specify a value from **0** through **86400** seconds. A value of 0 seconds causes the RPM test to stop after one iteration. The default value is 1.

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | 578

Examples: Configuring BGP Neighbor Discovery on MX, M, T and PTX Series Routers With RPM | 631

Examples: Configuring BGP Neighbor Discovery on MX, M, T and PTX Series Routers With RPM

Configure BGP neighbor discovery with RPM for all logical systems and all routing instances:

```
[edit services rpm]
bgp {
    probe-type icmp-ping;
    probe-count 5;
    probe-interval 1;
    test-interval 60;
    history-size 10;
    data-size 255;
    data-fill 0123456789;
}
```

Configure BGP neighbor discovery with RPM for only the following logical systems and routing instances: **LS1/RI1**, **LS1/RI2**, **LS2**, and **RI3**:

```
[edit services rpm]
bgp {
    probe-type icmp-ping;
    probe-count 5;
    probe-interval 1;
    test-interval 60;
    history-size 10;
    data-size 255;
    data-fill 0123456789;
    logical-system {
        LS1 {
            routing-instances {
                RI1;
                RI2;
            }
        }
```

```
        LS2;
    }
    routing-instance {
        RI3;
    }
}
```

NOTE: The **logical-system** statement is not supported on PTX Series Packet Transport routers.

Configure BGP neighbor discovery with RPM for only the default logical system and default routing instance:

```
[edit services rpm]
bgp {
    probe-type icmp-ping;
    probe-count 5;
    probe-interval 1;
    test-interval 60;
    history-size 10;
    data-size 255;
    data-fill 0123456789;
    logical-system {
        null {
            routing-instances {
                default;
            }
        }
    }
}
```

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | **578**

Configuring BGP Neighbor Discovery Through RPM | **629**

Tracing RPM Operations on MX, M, T and ACX Series Routers

Tracing operations track all RPM operations and record them in a log file. The logged error descriptions provide detailed information to help you solve problems faster.

By default, no events are traced. If you include the **traceoptions** statement at the **[edit services rpm]** hierarchy level, the default tracing behavior is the following:

- Important events are logged in a file called **rmopd** located in the **/var/log** directory.
- When the log file reaches 128 kilobytes (KB), it is renamed **rmopd.0**, then **rmopd.1**, and so on, until there are three trace files. Then the oldest trace file (**rmopd.2**) is overwritten. (For more information about how log files are created, see the *Junos OS System Log Messages Reference*.)
- Log files can be accessed only by the user who configures the tracing operation.

You can change this default behavior by using the **traceoptions** statements. Changing the defaults is described in the following sections:

Configuring the RPM Log File Name

By default, the name of the file that records RPM trace output is **rmopd**. To specify a different file name:

```
[edit services rpm traceoptions]
user @host set file filename
```

Configuring the Number and Size of RPM Log Files

To configure the limits on the number and size of RPM trace files:

```
[edit services rpm traceoptions]
user@host set file filename files number size size
```

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).

For example, set the maximum file size to 2 MB, and the maximum number of files to 20 for a log file named **rpmtrace**:

```
[edit services rpm traceoptions]
user@host set file rpmtrace files 20 size 2MB
```

When the **rpmtrace** file reaches 2 MB, it is renamed **rpmtrace.0**, and a new file called **rpmtrace** is created. When the new **rpmtrace** reaches 2 MB, **rpmtrace.0** is renamed **rpmtrace.1** and **rpmtrace** is renamed **rpmtrace.0**. This process repeats until there are 20 trace files. Then the oldest file (**rpmtrace.19**) is overwritten by **rpmtrace.18**.

Configuring Access to the Log File

By default, log files can be accessed only by the user who configures the tracing operation.

To specify that any user can read all log files:

```
[edit services rpm traceoptions]
user@host set file filename world-readable
```

To explicitly set the default behavior:

```
[edit services rpm traceoptions]
user@host set file filename no-world-readable
```

Configuring a Regular Expression for Lines to Be Logged

By default, the trace operation output includes all lines relevant to the logged events.

To refine the output by specifying a regular expression (regex) to be matched:

```
[edit services rpm traceoptions]
user@host set file filename match regular-expression
```

Configuring the Trace Operations

By default, if the **traceoptions** configuration is present, only important events are logged. You can configure the trace operations to be logged by including the following statements at the **[edit services rpm traceoptions]** hierarchy level:

```
flag {
    all;
    configuration;
    error;
    ipc;
    ppm;
    statistics
}
```

Table 101 on page 635 describes the meaning of the RPM tracing flags.

Table 101: RPM Tracing Flags

Flag	Description	Default Setting
all	Trace all operations.	Off
configuration	Trace configuration events.	Off
error	Trace events related to catastrophic errors in daemon.	Off
ipc	Trace IPC events.	Off
ppm	Trace ppm events.	Off
statistics	Trace statistics.	Off

SEE ALSO

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | 578

Examples: Configuring Real-Time Performance Monitoring on MX, M, T and PTX Series Routers

Configure an RPM instance identified by the probe name **probe1** and the test name **test1**:

```
[edit services rpm]
```

```
probe probe1{
   test test1 {
      dscp-code-points 001111;
      probe-interval 1;
      probe-type icmp-ping;
      target address 172.17.20.182;
      test-interval 20;
      thresholds rtt 10;
      traps rtt-exceeded;
   }
}
probe-server {
   tcp {
      destination-interface lt-0/0/0.0
      port 50000;
   }
   udp {
      destination-interface lt-0/0/0.0
      port 50001;
   }
}
probe-limit 200;
```

Configure packet classification, using **lt-** interfaces to send the probe packets to a logical tunnel input interface. By sending the packet to the logical tunnel interface, you can configure regular and multifield classifiers, firewall filters, and header rewriting for the probe packets. To use the existing tunnel framework, the **dlci** and **encapsulation** statements must be configured.

```
[edit services rpm]
probe p1 {
   test t1 {
      probe-type icmp-ping;
      target address 10.8.4.1;
      probe-count 10;
      probe-interval 10;
      test-interval 10;
      source-address 10.8.4.2;
      dscp-code-points ef;
      data-size 100;
      destination-interface lt-0/0/0.0;
   }
}
[edit interfaces]
lt-0/0/0 {
```

```
    unit 0 {
        encapsulation frame-relay;
        dlci 10;
        peer-unit 1;
        family inet;
    }
    unit 1 {
        encapsulation frame-relay;
        dlci 10;
        peer-unit 0;
        family inet;
    }
}
[edit class-of-service]
interfaces {
    lt-0/0/0 {
        unit 1 {
            classifiers {
                dscp default;
            }
        }
    }
}
```

Configure an input filter on the interface on which the RPM probes are received. This filter enables prioritization of the received RPM packets, separating them from the regular data packets received on the same interface.

```
[edit firewall]
filter recos {
    term recos {
        from {
            source-address {
                10.8.4.1/32;
            }
            destination-address {
                10.8.4.2/32;
            }
        }
        then {
            loss-priority high;
            forwarding-class network-control;
        }
    }
```

```
}
[edit interfaces]
fe-5/0/0 {
    unit 0 {
        family inet {
            filter {
                input recos;
            }
            address 10.8.4.2/24;
        }
    }
}
```

Configure an RPM instance and enable RPM for the extension-provider packages on the adaptive services interface:

```
[edit services rpm]
probe probe1{
    test test1 {
        data-size 1024;
        data-fill 0;
        destination-interface ms-1/2/0.10;
        dscp-code-points 001111;
        probe-count 10;
        probe-interval 1;
        probe-type icmp-ping;
        target address 172.17.20.182;
        test-interval 20;
        thresholds rtt 10;
        traps rtt-exceeded;
    }
}
[edit interfaces]
ms-1/2/0 {
    unit 0 {
        family inet;
    }
    unit 10 {
        rpm client;
        family inet {
            address 192.0.2.1/32;
        }
}
[edit chassis]
```

```
fpc 1 {
    pic 2 {
        adaptive-services {
            service-package {
                extension-provider {
                    control-cores 1;
                    data-cores 1;
                    object-cache-size 512;
                    policy-db-size 64;
                    package jservices-rpm;
                    syslog {
                        daemon any;
                    }
                }
            }
        }
    }
}
```

NOTE: TWAMP is not supported on PTX Series Packet Transport routers.

Configure the minimum statements necessary to enable TWAMP:

```
[edit services]
rpm {
    twamp {
        server {
            authentication-mode none;
            port 10000; # Twamp server's listening port
            client-list LIST-1 { # LIST-1 is the name of the client-list. Multiple lists can be configured.
                address {
                    198.51.100.2/30; # IP address of the control client.
                }
            }
        }
    }
[edit interfaces sp-5/0/0]
unit 0 {
    family inet;
}
unit 10 {
```

```
    rpm {
        twamp-server; # You must configure a separate logical interface on the service PIC interface for the TWAMP
            server.
    }
    family inet {
        address 203.0.113.50/32; # This address must be a host address with a 32-bit mask.
    }
}
[edit chassis]
fpc 5 {
    pic 0 {
        adaptive-services {
            service-package layer-2;      # Configure the service PIC to run in Layer 2 mode.
        }
    }
}
```

Configure additional TWAMP settings:

```
[edit services]
rpm {
    twamp {
        server {
            maximum-sessions 5;
            maximum-sessions-per-connection 2;
            maximum-connections 3;
            maximum-connections-per-client 1;
            port 10000;
            server-inactivity-timeout ;
            client-list LIST-1 {
                address {
                    198.51.100.2/30;
                }
            }
        }
    }
}
```

RELATED DOCUMENTATION

Understanding Using Probes for Real-Time Performance Monitoring on M, T, PTX and MX Series Routers | 578

Enabling RPM on MX, M and T Series Routers and SRX Firewalls for the Services SDK

Real-time performance monitoring (RPM), which has been supported on the adaptive services interface, is now supported by the Services SDK. RPM is supported on all platforms and service PICs that support the Services SDK.

To enable RPM for the Junos OS extension-provider package on the adaptive services interface, configure the **object-cache-size**, **policy-db-size**, and **package** statements at the **[edit chassis fpc *slot-number* pic *pic-number* adaptive-services service-package extension-provider]** hierarchy level. For the extension-provider package, ***package-name*** in the **package *package-name*** statement is **jservices-rpm**.

For more information about the extension-provider package, see the *SDK Applications Configuration Guide and Command Reference*.

The following example shows how to enable RPM for the extension-provider package on the adaptive services interface:

```
chassis fpc 1 {
   pic 2 {
      adaptive-services {
         service-package {
            extension-provider {
               control-cores 1;
               data-cores 1;
               object-cache-size 512;
               policy-db-size 64;
               package jservices-rpm;
               syslog daemon any;
            }
         }
      }
   }
}
```

RELATED DOCUMENTATION

CHAPTER 15

Managing License Server for Throughput Data Export

IN THIS CHAPTER

License Server Management for Throughput Data Export on MX Series Routers for NAT, Firewall, and Inline Flow Monitoring Services

To support our transition to software defined networking (SDN), Juniper Networks supports the Software Business Model Transformation, which includes new licensing, pricing, and branding strategies that make it easier for users to extract value from Juniper software solutions. This value of this approach is known as the Juniper Software Advantage (JSA), which provides the following benefits:

- Simple—Simple to buy, use, and manage rights
- Repeatable—License models which facilitates repeatable use among multiple hardware platforms and usage scenarios.
- Measurable—License fees based on easy to measure usage

Although the licensing of JSA products is trust-based, Juniper Networks might periodically audit the usage of its products. License Measurement Tool (LMT) is a technique that is used to compute the usage of individual Network Edge Products under JSA. MX Series routers need to define the mechanism for updating the LMT tool with information such as per-service throughput. For example, for services such as carrier-grade NAT and inline flow monitoring, the router needs to calculate per service throughput and update it in LMT.

On MX Series routers, the Routing Engine periodically sends query messages to every Service PIC on which the service, for which throughput collection is being performed, is configured to run. This polling is performed for all the services for which throughput measurement is enabled. Service PICs, upon receiving the query for a particular service, reply with the throughput measured during the last query interval, for that service. If a service PIC hosts multiple services, the Routing Engine sends separate throughput queries to that service PIC for all the services. If a service is configured on multiple services PICs, the Routing Engine aggregates the throughput values received from all of them and exports the aggregated throughput

to the log collector in the predefined log format. The LMT application analyze these values from log collector, performs aggregation on values collected from all routers, and displays them in the LMT application.

You can configure the capability to transmit the throughput details per service for the Junos Address Aware (carrier-grade NAT) and Junos Traffic Vision (previously known as Jflow) in the last time interval to an external log collector. The default time interval at which the throughput data is sent is 300 seconds, which you can configure to suit your network needs. Multiple instances of the same service running on different PICs within a router are supported. If the same service is running on different PICs within a router, the router transmits the consolidated final throughput to the log collector or server. This functionality is supported on MX Series routers with MS-MCPs and MS-MICs, and also in the MX Series Virtual Chassis configuration. To configure the license server properties for throughput data to be transmitted for the defined services, such as NAT or stateful firewall, from the service PIC on the router to the external log collector, include the license-server statement at the [edit] hierarchy level. To specify the IP address of the license log server, include the **ip-address *address*** statement at the **[edit license-server]** hierarchy level. To configure the frequency of transmission of throughput date, include the **log-interval *seconds*** statement at the [edit license-server] hierarchy level. To specify the services for which throughput data collection must be performed, include the **services (jflow | cgnat | firewall)** statement at the **[edit license-server]** hierarchy level.

Throughput Measurement and Export

Throughput is defined as: “The network traffic throughput processed by juniper software in a second. It is represented as Mb/Sec (Megabits per second) or GB/sec (Gigabits per second). Throughput is measured as the 95th percentile of all the peaks measured in a quarter.” Service PICs keep track of the amount of data (in bits) processed by the various service plugins running on them. When a throughput query arrives from the Routing Engine, for a particular service, the Service PIC returns the value D/T mbps, in its reply, where:

- D is the amount of data (megabits) processed by that service since the previous query was received. If the query interval happens to be 300 seconds, for example, then D refers to the amount of data that was processed during the last 300 second interval. If the current query happens to be the very first query, for a particular service, then D represents the cumulative data bits processed so far, by that service.
- T is the time (seconds) that elapsed since the previous query was received. This is the query interval configured using the CLI interface. If the current query happens to be the very first query, for a particular service, then T represents the time that elapsed since that service started processing packets. For all subsequent queries, T equals the query interval.

The Routing Engine aggregates the throughput measured (in mbps) across all the Service PICs on which a particular Service is configured and exports it to the Log collector which performs the 95th percentile calculation.

Guidelines for Configuring an MX Series Router to Transmit Per-Service Throughput to an External Log Collector

Observe the following guidelines while configuring this functionality on MX Series routers with MS-MPCs and MS-MICs:

- If the syslog server is unreachable, the router cannot send information to the log collector.
- After a graceful Routing Engine switchover (GRES) procedure, the newly functioning active Routing Engine starts sending the data to the server after the configured time interval, which is similar to a reset operation. The time elapsed in the active interval and data before GRES are not preserved.
- The time range must be from 60 through 86400 seconds (24 hours).
- If the timer is not configured, the default value of 300 seconds is assumed.
- The throughput data can be sent only if a service is up and running.
- Only maximum throughput is transmitted for the last 300 seconds or the configured time interval.
- The throughput value must not be less than zero to enable transmission. The data is sent based on the timezone of the router.
- An acknowledgment mechanism for data sent to the log collector is not supported. The router does not receive any acknowledgement regarding whether the data is already written into the log collector.
- The router does not maintain throughput data beyond the configurable time interval.
- No mechanisms exist to track if the log collector is successfully receiving the sent data or if the log server is reachable.
- The time interval and log collector are common for all the services; you cannot configure a different period for collection of logs for each service or a different log collector for each service.
- You cannot clear the system throughput value using a CLI command. It is assumed that the throughput value is not cleared or changed from outside. Throughput must be calculated internally by services and must not be manually modified by a CLI.
- SNMP support for these values is not available.
- The log collector performs a 95 percentile calculation of throughput data. Syslogs are sent even in scaled system conditions to the log collector for the throughput data related to the configured services.
- The following is the format of the syslogs configured to be sent at the prescribed frequency:

```
<Date> <Time> < time-zone> <Router_name> <Service_name> <Throughput_value>
Throughput = <Unit_Mbps/Gbps> in last <Time_Interval>
```

An example is as follows:

```
Jan  8 08:49:57  America/Adak deuterium  CGNAT Throughput = 1500000 Mbps in last
 300Sec
```

CHAPTER 16

Testing the Performance of Network Devices Using RFC 2544-Based Benchmarking

IN THIS CHAPTER

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers

RFC2544 defines a series of tests that can be used to describe the performance characteristics of a network-interconnecting device, such as a router, and outlines specific formats to report the results of the tests. These tests can be used to benchmark interconnected network devices and devise a guideline or a measurement pattern to analyze the health and efficiency of the network devices. These tests are the standard benchmarking tests for Ethernet networks and are known as RFC2544-based benchmarking tests. These tests measure throughput, latency, frame loss rate, and bursty frames. The test methodology enables you to define various parameters such as different frame sizes to be examined (64, 128, 256, 512, 1024, 1280, and 1518 bytes), the test time for each test iteration (10 seconds through 1,728,000 seconds), and the frame format (UDP-over-IP).

NOTE: MX Series routers support only the reflector function in RFC2544-based benchmarking tests.

NOTE: RFC2544-based benchmarking tests support only UDP over IPv4 test traffic (unicast).

An RFC2544-based benchmarking test is performed by transmitting test packets from a device that functions as the generator or the initiator (which is also called the originator). These packets are sent to a device that functions as a reflector, which receives and returns the packets to the initiator.

MX80 and MX104 Universal Routing Platforms and, starting from Junos OS Release 16.1, MX240, MX480, and MX960 Universal Routing Platforms with MPC1 (MX-MPC1-3D), MPC2 (MX-MPC2-3D), and the 16-port 10-Gigabit Ethernet MPC (MPC-3D-16XGE-SFP) support the reflector function and the corresponding benchmarking tests. Starting from Junos OS Release 17.1R1, the reflector function and the corresponding benchmarking tests are supported on MX Series routers with MPC3E (MX-MPC3E-3D), MPC3E-NG (MX-MPC3E-3D-NG), MPC4E (MPC4E-3D-32XGE-SFPP and MPC4E-3D-2CGE-8XGE), MPC5E (MPC5E-40G10G, MPC5EQ-40G10G, MPC5E-100G10G, and MPC5EQ-100G10G), and MPC6E (MX2K-MPC6E). Starting from Junos OS Release 15.1, MX104 Series routers also perform verification of signatures on the received test frames. By default, when the MX104 Series router receives a test packet that does not have the signature pattern, the packet is dropped. If you generate test traffic using a third-party vendor tool instead of an ACX Series router, you can disable signature verification. To disable signature verification, use the **disable-signature-check** command.

NOTE: To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 Series routers with MPC1, MPC2, and the 16-port 10-Gigabit Ethernet MPC, see “Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers” on page 666.

The RFC2544-based benchmarking test methodology assesses different parameters that are defined in service-level agreements (SLAs). By measuring the performance availability, transmission delay, link bursts, and service integrity, a carrier provider can certify that the working parameters of the deployed Ethernet circuit comply with the SLA and other defined policies.

Table 102 on page 649 describes the different network topologies in which the benchmarking test is supported.

Table 102: Supported Network Topologies for RFC2544 Benchmarking Tests

Service Type	Traffic Direction	Mode	Initial Release on MX104 Series Routers	Initial Release on MX240, MX480, and MX960 Series Routers	Whether the Benchmarking Test Is Supported
E-Line (family **bridge**)	(UNI) Egress (UNI) Ingress	Port Port, VLAN	14.2R1 (E-Line family **bridge**) 17.1R1	16.1R1 17.1R1	Supported
E-LAN (family **bridge** and family **vpls**)	(UNI) Egress (UNI) Ingress	Port Port, VLAN	14.2R1 (E-LAN family **bridge)** 15.1R1 (E-LAN family **vpls**) 17.1R1	16.1R1 17.1R1	Supported
E-Line (family **ccc**)	Ingress Egress	Port Port, VLAN	13.3R1 (E-Line pseudowire)	16.1R1	Supported
IP Services (family **inet**)	NNI	Port Port, VLAN	13.3R1	16.1R1	Supported

NOTE: You can configure a total of four simultaneous active reflection sessions. The four active reflection sessions can be of the same type or can be a combination of the different types of reflection sessions. For instance, you can configure either four IPv4 reflection sessions or one session each for pseudowire reflection, VPLS reflection, Layer 2 reflection, and IPv4 reflection. The maximum reflection bandwidth supported is 4 Gbps in a standalone test condition.

Table 103 on page 650 lists the interfaces and the reflection type on which the benchmarking tests are supported.

Table 103: Supported Interfaces for RFC2544 Benchmarking Tests

Type of Reflection	Gigabit Interfaces (ge)	Aggregated Interfaces (ae)	10G Interfaces (xe)	Pseudo Interfaces (irb, lt, vt, lo0, and others)
IPv4	Yes	No	No	No
Pseudowire ingress	Yes	No	No	No
Pseudowire egress	Yes	Yes (starting in Junos OS Release 15.1)	Yes (starting in Junos OS Release 15.1)	No
Layer 2 bridge	Yes	Yes	Yes	No
Layer 2 VPLS	Yes	Yes	Yes	No

All active RFC2544-based benchmarking tests are stopped when any of the following events takes place:

- System events such as Packet Forwarding Engine restarts, Routing Engine restarts, and so on.
- Test interface change events such as deactivation and reactivation of the interface, disabling and enabling of the interface, child link events for aggregated interfaces and so on.

  After the benchmarking tests are stopped, the test states of the tests are removed and the user can restart the same test. Other ongoing tests on other interfaces are not interrupted.

> **NOTE:** RFC2544-based benchmarking tests are not supported during unified in-service software upgrade (ISSU) and graceful Routing Engine switchover (GRES).

Release History Table

Release	Description
17.1R1	Starting from Junos OS Release 17.1R1, the reflector function and the corresponding benchmarking tests are supported on MX Series routers with MPC3E (MX-MPC3E-3D), MPC3E-NG (MX-MPC3E-3D-NG), MPC4E (MPC4E-3D-32XGE-SFPP and MPC4E-3D-2CGE-8XGE), MPC5E (MPC5E-40G10G, MPC5EQ-40G10G, MPC5E-100G10G, and MPC5EQ-100G10G), and MPC6E (MX2K-MPC6E).
16.1	MX80 and MX104 Universal Routing Platforms and, starting from Junos OS Release 16.1, MX240, MX480, and MX960 Universal Routing Platforms with MPC1 (MX-MPC1-3D), MPC2 (MX-MPC2-3D), and the 16-port 10-Gigabit Ethernet MPC (MPC-3D-16XGE-SFP) support the reflector function and the corresponding benchmarking tests.
15.1	Starting from Junos OS Release 15.1, MX104 Series routers also perform verification of signatures on the received test frames.

RELATED DOCUMENTATION

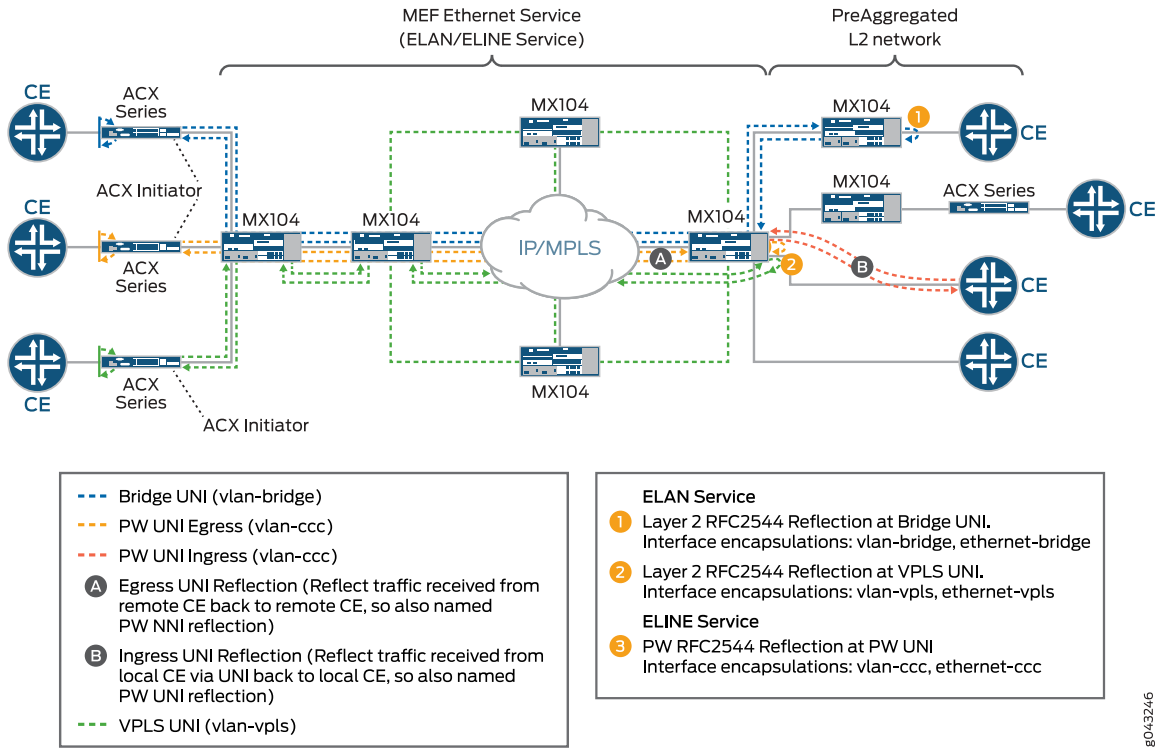
- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Supported RFC2544-Based Benchmarking Statements on MX Series Routers | **657**
- Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers | **666**

Understanding RFC2544-Based Benchmarking Tests for E-LAN and E-Line Services on MX Series Routers

NOTE: MX Series routers support only the reflector function in RFC2544-based benchmarking tests.

The Metro Ethernet Forum (MEF) defines two Ethernet service types—E-LAN and E-Line—and specifies the associated service attributes and parameters. These services can be supported within the Metro Ethernet Network (MEN) and also supported over different transport technologies such as SONET, MPLS, and so on. Juniper Networks ACX Series routers, MX80, MX104 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-Gigabit Ethernet MPC provide support for Layer 2 E-LAN and E-Line services reflection. Figure 50 on page 653 shows a sample topology for the E-LAN and E-Line reflection supported on MX104 Series routers.

Figure 50: E-LAN And E-Line Reflection in a metro Solution



Starting in Junos OS Release 15.1, MX104 Series routers support RFC2544-based benchmarking tests for Layer 2 reflection (E-Line service) by using pseudowires (Layer 2 circuit and L2VPN). Starting in Junos OS Release 16.1, MX80 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-gigabit Ethernet MPC support RFC2544-based benchmarking tests for Layer 2 reflection (E-Line service). E-Line provides transparent data transport. You can configure RFC2544-based benchmarking tests for both ingress and egress direction on the customer edge (CE) facing interface of family type **CCC** for an Ethernet pseudowire.

NOTE: To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 Series routers with MPC1, MPC2, and the 16-port 10-Gigabit Ethernet MPC, see "Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers" on page 666.

Starting in Junos OS Release 15.1, MX104 routers support RFC2544-based benchmarking tests for Layer 2 reflection (E-LAN service) by using VPLS and basic bridge domains. In Junos OS Release 14.2 and earlier, only basic bridge domains are used. Starting in Junos OS Release 16.1, MX80 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-gigabit Ethernet MPC support RFC2544-based benchmarking tests for Layer 2 reflection (E-LAN service) by using VPLS and basic bridge domains. VPLS enables geographically dispersed sites to share an Ethernet broadcast domain by connecting sites across an MPLS network. All sites appear to be in the same Ethernet LAN though traffic travels across the MPLS network. Both LDP-based VPLS and BGP-based VPLS are supported. RFC2544-based benchmarking and performance measurement testing for Layer 2 E-LAN services (**bridge**/ **VPLS**) is supported on unicast traffic in egress direction only.

During the benchmarking tests, the initiator or generator transmits a test packet (unicast) to a reflector. The reflector receives and reflects the test packet back to the initiator. The test packet is an UDP-over-IP packet with a source and destination MAC address.

In a E-LAN service, the Layer 2 traffic reflection session is identified by the source MAC address, the destination MAC address, and the egress interface (logical interface). By default, RFC2544-based benchmarking tests are performed when there is no other service traffic. This mode of operation is known as out-of-service mode. The default service mode for the reflecting egress interface for an E-LAN service is also out-of-service mode. In out-of-service mode, while the test is running, all the data traffic (other than test traffic) sent to and from the test interface under test is interrupted. If the test is activated on a logical interface, all the traffic sent to and from the logical interface is interrupted. However, if there are other logical interfaces on the UNI port, the traffic sent to and from those logical interfaces is not interrupted. Control protocol peering is not interrupted whereas pass through control protocol packets such as end-to-end CFM sessions are interrupted. If you do not want the control protocol packets interrupted, you can configure the E-LAN service mode as in-service mode. In the in-service mode, while the test is running, the rest of the data traffic flow sent to and from the UNI port under test on the service is not interrupted. Both peering and pass through control protocols are not interrupted.

In an E-Line service, the reflection session is identified by the egress interface which is the logical interface. On activation of reflection on a logical interface, the traffic received on the logical interface is reflected. You can specify the type of traffic you want reflected by specifying the EtherType (specifies the protocol transported). If you do not specify the EtherType, all traffic is reflected. System does not explicitly block other traffic on the test interface during E-line service. You can block non-test traffic using firewall filters.

By default, for E-LAN services, the reflector swaps MAC addresses. The reflector swaps the source and destination MAC addresses and sends the packet back to the initiator. By default, for E-Line services, the

reflector does not swap MAC addresses. Table 104 on page 655 describes the MAC address swapping behavior for the service types.

Table 104: MAC Address Swapping Behavior for E-LAN and E-Line Services

Family	Direction	Default Behavior	User-configurable
bridge	Egress	MAC address swap (E-LAN service type) No MAC address swap (E-Line service type)	No Yes
vpls	Egress	MAC address swap (E-LAN service type)	No
ccc	Egress	No MAC address swap	Yes (starting in Junos OS Release 15.1)
	Ingress	MAC address swap	No

By default, the IP addresses and UDP ports are not modified. Optionally, you can configure the reflector to swap the source and destination IP address and the source and destination UDP ports.

You can configure an ACX Series router to operate as an initiator as well as a reflector. The MX104 Series router can be configured to operate only as a reflector.

Starting in Junos OS Release 15.1, MX104 Series routers support the specification of the protocol transported in the Ethernet frame. Starting in Junos OS Release 16.1, MX80 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-gigabit Ethernet MPC also support the specification of the protocol transported in the Ethernet frame. To specify the EtherType (specifies the protocol transported) used for reflection of the test frames, use the **reflect-etype** command. If you do not specify the EtherType, all EtherTypes are reflected.

NOTE: The maximum reflection bandwidth supported is 4 Gbps. Because RFC2544 reflection shares system bandwidth with other loopback services such as tunnel services, you must manage the sharing of bandwidth for performing RFC2544-based performance tests.

NOTE: RFC2544-based benchmarking tests are not supported during unified in-service software upgrade (ISSU) and graceful Routing Engine switchover (GRES).

Release History Table

Release	Description
16.1	Starting in Junos OS Release 15.1, MX104 Series routers support the specification of the protocol transported in the Ethernet frame.
15.1	Starting in Junos OS Release 15.1, MX104 Series routers support RFC2544-based benchmarking tests for Layer 2 reflection (E-Line service) by using pseudowires (Layer 2 circuit and L2VPN).
15.1	Starting in Junos OS Release 15.1, MX104 routers support RFC2544-based benchmarking tests for Layer 2 reflection (E-LAN service) by using VPLS and basic bridge domains.

RELATED DOCUMENTATION

- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- Supported RFC2544-Based Benchmarking Statements on MX Series Routers | **657**
- Example: Configuring RFC2544-Based Benchmarking Tests on an MX104 Router for Layer 2 E-LAN Services in Bridge Domains | **701**
- disable-signature-check | **926**
- reflect-etype | **1104**

Supported RFC2544-Based Benchmarking Statements on MX Series Routers

Table 105 on page 658 lists the reflector-specific configuration statements that are supported on the MX104 Series routers. Note that en dash (–) specified in the Initial Release on MX Series routers column denotes that the command is not supported.

Table 105: Supported RFC2544-Based Benchmarking Reflector Statements on MX Series

Statement	Options	Initial Release on MX104 Series Routers	Intial Release on MX240, MX40, MX960 Series Routers
destination-ipv4-address	–	13.3R1	16.1R1
destination-mac-address	–	14.2R1	16.1R1
destination-udp-port	–	13.3R1	16.1R1
direction	(**egress** | **ingress**)	13.3R1	16.1R1
disable-signature-check	–	15.1R1	16.1R1
family	(**ccc** | **inet**) (**bridge** | **ccc** | **inet**) (**vpls**)	13.3R1 14.2R1 15.1R1	16.1R1
in-service	–	14.2R1	16.1R1
ip-swap	–	14.2R1	16.1R1
mode	**reflect**	13.3R1	16.1R1
reflect-etype	–	15.1R1	16.1R1
reflect-mode	(**mac-swap** | **no-mac-swap**)	14.2R1	16.1R1
service-type	(**eline** | **elan**)	14.2R1	16.1R1
source-ipv4-address	–	13.3R1	16.1R1
source-mac-address	–	14.2R1	16.1R1
source-udp-port	–	13.3R1	16.1R1

Table 105: Supported RFC2544-Based Benchmarking Reflector Statements on MX Series *(continued)*

Statement	Options	Initial Release on MX104 Series Routers	Intial Release on MX240, MX40, MX960 Series Routers
test-interface	–	13.3R1	16.1R1
udp-tcp-port-swap	–	14.2R1	16.1R1

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

Configuring an RFC 2544-Based Benchmarking Test | **659**

Example: Configuring RFC2544-Based Benchmarking Tests on an MX104 Router for Layer 2 E-LAN Services in Bridge Domains | **701**

Configuring an RFC 2544-Based Benchmarking Test

IN THIS SECTION

You can configure a benchmarking test to detect and measure performance attributes, such as throughput, latency, frame loss, and bursty or back-to-back frames, of network devices. RFC 2544-based benchmarking test is performed by transmitting test packets from a device that functions as the generator or the initiator. These packets are sent to a device that functions as the reflector, which receives and returns the packets back to the initiator.

> **NOTE:** The test configuration is applied only when you start the test. If you update the test configuration during the test, you have to start the test again for the updated configuration to take effect.

You must configure a test profile and reference the test profile in a unique test name that defines the parameters for the test to be performed on a certain device. However, the test profile is required when the test mode is configured as initiation and termination. The **test-profile** parameter is disregarded when the test mode is configured as reflection. MX Series routers support only the reflection function in the RFC 2544-based benchmarking tests. A reflection service does not use the parameters specified in the test profile.

> **NOTE:** To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 Series routers with MPC1, MPC2, and the 16-port 10-Gigabit Ethernet MPC, see "Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers" on page 666.

The following topics describe how to configure a test name for an RFC 2544-based benchmarking test on an MX Series router for Layer 3 IPv4, Ethernet pseudowire, and Layer 2 bridge networks:

Configuring a Test Name for an RFC 2544-Based Benchmarking Test for a IPv4 Network

You can configure a test name by including the **test-name *test-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level. In the test name, you can configure attributes of the test iteration, such as the address family (type of service, IPv4 or Ethernet), the logical interface, and test duration that are used for a benchmarking test to be run.

To configure a test name and define its attributes for an IPv4 network:

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure a instance.

```
[edit services]
user@host# edit rpm
```

3. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

4. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

5. Specify the test mode for the packets that are sent during the benchmarking test. The **reflect** option causes the test frames to be reflected on the IPv4 network.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

6. Configure the address type family for the benchmarking test. The **inet** option indicates that the test is run on an IPv4 service.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

7. Configure the destination IPv4 address for the test packets. This parameter is required only if you configure IPv4 family **inet**. If you do not configure the destination IPv4 address, the default value of 192.168.1.20 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-ipv4-address address
```

8. Specify the UDP port of the destination to be used in the UDP header for the generated frames. If you do not specify the UDP port, the default value of 4041 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-udp-port port-number
```

9. (Optional) Specify the source IPv4 address to be used in generated test frames. If you do not configure the source IPv4 address for **inet** family, the source address of the interface is used to transmit the test frames.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-ipv4-address address
```

10. Specify the UDP port of the source to be used in the UDP header for the generated frames. If you do not specify the UDP port, the default value of 4041 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-udp-port port-number
```

11. Specify the logical interface on which the RFC 2544-based benchmarking test is run. If you configure an **inet** family and the test mode to reflect the frames back on the sender from the other end, then the logical interface is used as the interface to enable the reflection service (reflection is performed on the packets entering the specified interface). If you not configure the logical interface for reflection test mode, then a lookup is performed on the source IPv4 address to determine the interface that hosts the address.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface interface-name
```

Configuring a Test Name for an RFC 2544-Based Benchmarking Test for an Ethernet Pseudowire

You can configure a test name by including the **test-name *test-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level. In the test name, you can configure attributes of the test iteration, such as the address family (type of serviceIPv4 or Ethernet), the logical interface, and test duration, that are used for a benchmarking test to be run. The test name combined with the test profile represent a single real-time performance monitoring (RPM) configuration instance.

To configure a test name and define its attributes for an Ethernet Pseudowire:

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure an RPM service instance.

```
[edit services]
user@host# edit rpm
```

3. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

4. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

5. Specify the test mode for the packets that are sent during the benchmarking test. The **reflect** option causes the test frames to be reflected on the Ethernet pseudowire.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

6. Configure the address type family for the benchmarking test. The **ccc** option indicates that the test is run on a CCC or Ethernet pseudowire service.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

7. Specify the direction of the interface on which the test must be run. This parameter is valid only for a family. To enable the test to be run in the egress direction of the interface (network-to-network interface (NNI)), use the **egress** option. To enable the test to be run in the ingress direction of the interface (user-to-network interface (UNI)), use the **ingress** option.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction (egress | ingress)
```

8. (Optional) Specify the source IPv4 address to be used in generated test frames. If you do not configure the source IPv4 address for family, the default value of 192.168.1.10 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-ipv4-address address
```

9. Specify the logical interface on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface interface-name
```

Configuring a Test Name for an RFC 2544-Based Benchmarking Test for a Layer 2 E-LAN Service in Bridge Domain

You can configure a test name by including the **test-name *test-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level. In the test name, you can configure attributes of the test iteration, such as the address family (bridge), the logical interface, and test duration, that are used for a benchmarking test to be run. The test name combined with the test profile represent a single real-time performance monitoring (RPM) configuration instance.

To configure a test name and define its attributes for a layer 2 E-LAN service in Bridge domains:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   ```
   [edit]
   user@host# edit services
   ```

2. Configure an RPM service instance.

   ```
   [edit services]
   user@host# edit rpm
   ```

3. Configure an RFC 2544-based benchmarking test for the RPM instance.

   ```
   [edit services rpm]
   user@host# edit rfc2544-benchmarking
   ```

4. Define a name for the test—for example, l2b-test1. The test name identifier can be up to 32 characters in length.

   ```
   [edit services rpm rfc2544-benchmarking]
   user@host# edit tests test-name l2b-test1
   ```

5. Specify the source and destination MAC addresses of the test packet. Both these parameters are valid only for the bridge family.

```
[edit services rpm rfc2544-benchamrking tests test-name l2b-test1]
user@host# set source-mac-address address destination-mac-address address
```

6. Specify the service type under test. This parameter is applicable only for the bridge family.

```
[edit services rpm rfc2544-benchamrking tests test-name l2b-test1]
user@host# set service-type elan
```

7. Specify the test mode for the packets that are sent during the benchmarking test. The **reflect** option causes the test frames to be reflected over the Layer 2 bridge.

```
[edit services rpm rfc2544-benchmarking tests test-name l2b-test1]
user@host# set mode reflect
```

8. Configure the address type family for the benchmarking test. The **bridge** option indicates that the test is run on a E-LAN service over a bridge domain.

```
[edit services rpm rfc2544-benchmarking tests test-name l2b-test1]
user@host# set family bridge
```

9. Specify the direction of the interface on which the test must be run. This parameter is valid only for a family. To enable the test to be run in the egress direction of the interface (network-to-network interface (NNI)), use the **egress** option. To enable the test to be run in the ingress direction of the interface (user-to-network interface (UNI)), use the **ingress** option.

```
[edit services rpm rfc2544-benchmarking tests test-name l2b-test1]
user@host# set direction egress
```

10. Specify the logical interface on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name l2b-test1]
user@host# set test-interface interface-name
```

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647

Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers

NOTE: MX Series routers support only the reflector function in RFC2544-based benchmarking tests.

RFC2544 defines a series of tests that can be used to describe the performance characteristics of a network-interconnecting device, such as a router, and outlines specific formats to report the results of the tests. These tests can be used to benchmark interconnected network devices and devise a guideline or a measurement pattern to analyze the health and efficiency of the network devices. These tests are known as RFC2544-based benchmarking tests and are supported on MX80, MX104, MX240, MX480, MX960, and MX2010 routers with MPC1 (MX-MPC1-3D), MPC2 (MX-MPC2-3D), and the 16-port 10-Gigabit Ethernet MPC (MPC-3D-16XGE-SFP). Starting from Junos OS Release 17.1R1, the RFC2544-based benchmarking tests are supported on MX Series routers with MPC3E (MX-MPC3E-3D), MPC3E-NG (MX-MPC3E-3D-NG), MPC4E (MPC4E-3D-32XGE-SFPP and MPC4E-3D-2CGE-8XGE), MPC5E (MPC5E-40G10G, MPC5EQ-40G10G, MPC5E-100G10G, and MPC5EQ-100G10G), and MPC6E (MX2K-MPC6E).

NOTE: On MX104 and MX80 Series routers that have a single fixed FPC, this configuration is not required.

To enable support for RFC2544-based benchmarking tests on MX Series routers:

1. In configuration mode, go to the [**edit chassis fpc *fpc-slot-number***] hierarchy level.

```
[edit]
user@host# edit chassis fpc fpc-slot-number
```

2. Enable support for service level monitoring services and RFC-based benchmarking tests

```
[edit chassis fpc fpc-slot-number]
user@host# set slamon-services rfc2544
```

Release History Table

Release	Description
17.1R1	Starting from Junos OS Release 17.1R1, the RFC2544-based benchmarking tests are supported on MX Series routers with MPC3E (MX-MPC3E-3D), MPC3E-NG (MX-MPC3E-3D-NG), MPC4E (MPC4E-3D-32XGE-SFPP and MPC4E-3D-2CGE-8XGE), MPC5E (MPC5E-40G10G, MPC5EQ-40G10G, MPC5E-100G10G, and MPC5EQ-100G10G), and MPC6E (MX2K-MPC6E).

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

Configuring an RFC 2544-Based Benchmarking Test | **659**

slamon-services | **1141**

Example: Configuring an RFC 2544-Based Benchmarking Test on an MX104 Router for Layer 3 IPv4 Services

IN THIS SECTION

Requirements

> **NOTE:** MX Series routers support only the reflector function in RFC2544-based benchmarking tests. This example uses the MX104 3D Universal Edge Router as the reflector. You can also configure benchmarking tests on MX80 routers and on MX240, MX480, and MX960 routers with MPC1, MPC2, and 16-port 10-Gigabit Ethernet MPC from Junos OS Release 16.1 or later. To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 routers, see "Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers" on page 666.

This example uses the following hardware and software components:

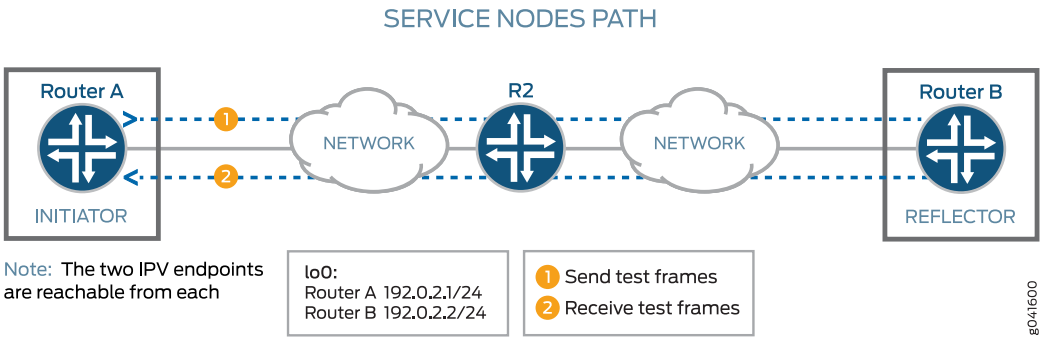
- An MX104 (reflector)
- An ACX Series router (initiator)
- Junos OS Release 13.3 or later

Overview

Consider a sample topology in which a router, Router A (ACX), functions as an initiator and terminator of the test frames for an RFC 2544-based benchmarking test. Router A is connected over a Layer 3 network to another router, Router B (MX104), which functions as a reflector to reflect back the test frames it receives from Router A. IPv4 is used for transmission of test frames over the Layer 3 network. This benchmarking test is used to compute the IPv4 service parameters between Router A and Router B. Logical interfaces on both the routers are configured with IPv4 addresses to measure the performance attributes, such as throughput, latency, frame loss, and bursty frames, of network devices for the IPv4 service.

Figure 51 on page 668 shows the sample topology to perform an RFC 2544 test for a Layer 3 IPv4 Service.

Figure 51: RFC 2544-Based Benchmarking Test for a Layer 3 IPv4 Service



Configuration

IN THIS SECTION

In this example, you configure the benchmarking test for a Layer 3 IPv4 service that is between interface ge-0/0/0 on Router A and interface ge-0/0/4 on Router B to detect and analyze the performance of the interconnecting routers.

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:

Configuring Benchmarking Test Parameters on Router A

```
set interfaces ge-0/0/0 unit 0 family inet address 192.0.2.1/24
set services rpm rfc2544-benchmarking profiles test-profile throughput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile throughput packet-size 64
set services rpm rfc2544-benchmarking profiles test-profile throughput test-duration 20m
set services rpm rfc2544-benchmarking profiles test-profile throughput bandwidth-kbps 500
set services rpm rfc2544-benchmarking tests test-name test1 test-profile throughput
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/0.1
set services rpm rfc2544-benchmarking tests test-name test1 mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name test1 family inet
set services rpm rfc2544-benchmarking tests test-name test1 dest-address 192.0.2.2
set services rpm rfc2544-benchmarking tests test-name test1 udp-port 4001
```

Configuring Benchmarking Test Parameters on Router B

```
set interfaces ge-0/0/4 unit 0 family inet address 192.0.2.2/24
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/4.1
set services rpm rfc2544-benchmarking tests test-name test1 mode reflect
set services rpm rfc2544-benchmarking tests test-name test1 family inet
set services rpm rfc2544-benchmarking tests test-name test1 dest-address 192.0.2.1
set services rpm rfc2544-benchmarking tests test-name test1 udp-port 4001
```

Configuring Benchmarking Test Parameters on Router A

Step-by-Step Procedure

The following requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router A:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

   ```
   [edit]
   user@host# edit interfaces
   ```

2. Configure the interface on which the test must be run.

   ```
   [edit interfaces]
   user@host# edit ge-0/0/0
   ```

3. Configure a logical unit and specify the protocol family.

   ```
   [edit interfaces ge-0/0/0]
   user@host# edit unit 0 family inet
   ```

4. Specify the address for the logical interface.

   ```
   [edit interfaces ge-0/0/0 unit 0 family inet]
   user@host# set address 192.0.2.1/24
   ```

5. Go to the top level of the configuration command mode.

   ```
   [edit interfaces ge-0/0/0 unit 0]
   user@host# top
   ```

6. In configuration mode, go to the **[edit services rpm rfc2544-benchmarking]** hierarchy level.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

7. Define a name for a test profile—for example, throughput.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile throughput
```

8. Configure the type of test to be performed as throughput.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type throughput
```

9. Specify the size of the test packet as 64 bytes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set packet-size 64
```

10. Specify the period for which the test is to be performed in hours, minutes, or seconds by specifying a number followed by the letter h (for hours), m (for minutes), or s (for seconds), respectively.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-duration 20m
```

11. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set bandwidth-kbps 500
```

12. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# up
```

13. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# up
```

14. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

15. Specify the name of the test profile—for example, throughput—to be associated with a particular test name.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile throughput
```

16. Specify the logical interface, ge-0/0/0.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/0.1
```

17. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode initiate-and-terminate
```

18. Configure the address type family, **inet**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

19. Configure the destination IPv4 address for the test packets.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dest-address 192.0.2.2
```

20. Specify the UDP port of the destination to be used in the UDP header for the generated frames as 4001.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set udp-port 4001
```

21. Start the benchmarking test on the initiator.

```
user@> test services rpm rfc2544-benchmarking test test1 start
```

After the test is successfully completed, it is automatically stopped at the initiator.

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router B:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/4
```

3. Configure a logical unit and specify the protocol family as **inet**.

```
[edit interfaces ge-0/0/4]
user@host# edit unit 0 family inet
```

4. Specify the address for the logical interface.

```
[edit interfaces ge-0/0/4 unit 0 family inet]
user@host# set address 192.0.2.2/24
```

5. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# top
```

6. In configuration mode, go to the **[edit services rpm rfc2544-benchmarking]** hierarchy level.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

7. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

8. Specify the logical interface, ge-0/0/4.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/4.1
```

9. Specify **reflect** as the test mode for the packets that are sent during the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

10. Configure the address type family, **inet**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

11. Configure the destination IPv4 address for the test packets as 192.0.2.1.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dest-address 192.0.2.1
```

12. Specify the UDP port of the destination to be used in the UDP header for the generated frames as 4001.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set udp-port 4001
```

13. Start the benchmarking test on the reflector.

```
user@host> test services rpm rfc2544-benchmarking test test1 start
```

After the test is successfully completed at the initiator, you can stop the test at the reflector by entering the **test services rpm rfc2544-benchmarking test test1** command.

Results

In configuration mode, confirm your configuration on Router A and Router B by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Benchmarking Test Parameters on Router A:

```
[edit interfaces]
  ge-0/0/0 {
      unit 0 {
          family inet {
              address 192.0.2.1/24;
          }
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      profiles {
          test-profile throughput {
              test-type throughput
              packet-size 64;
              test-duration 20m;
              bandwidth-kbps 500;
          }
      }

      tests {
          test-name test1 {
              test-profile throughput;
              interface ge-0/0/0.1;
```

```
            mode initiate,terminate;
            family inet;
            dest-address 192.0.2.2
            udp-port 4001;
        }
    }
}
```

Benchmarking Test Parameters on Router B:

```
[edit interfaces]
  ge-0/0/4 {
      unit 0 {
          family inet {
              address 192.0.2.2/24;
          }
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
      tests {
          test-name test1 {
              interface ge-0/0/4.1;
              mode reflect;
              family inet;
              dest-address 192.0.2.1;
              udp-port 4001;
          }
      }
  }
```

After you have configured the device, enter the **commit** command in configuration mode.

Verifying the Results of the Benchmarking Test for Layer 3 IPv4 Services

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between Router A and Router B.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between Router A and Router B.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647

Configuring an RFC 2544-Based Benchmarking Test | 659

Example: Configuring an RFC 2544-Based Benchmarking Test on an MX104 Router for UNI Direction of Ethernet Pseudowires

IN THIS SECTION

This example shows how to configure the benchmarking test for the user-to-network interface (UNI) direction of an Ethernet pseudowire service.

Requirements

> **NOTE:** MX Series routers support only the reflector function in RFC2544-based benchmarking tests. This example uses the MX104 3D Universal Edge Router as the reflector. You can also configure benchmarking tests on MX80 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-gigabit Ethernet MPC from Junos OS Release 16.1 or later. To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 Series routers, see "Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers" on page 666.

This example uses the following hardware and software components:

- An MX104 (reflector)
- An ACX Series router (initiator)
- Junos OS Release 13.3 or later

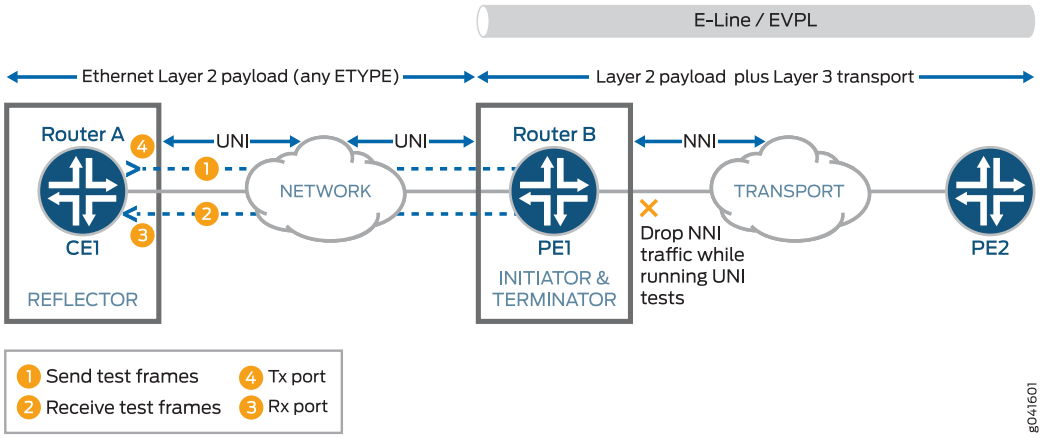
Overview

Consider a sample topology in which a router, Router A (MX104), functions as a reflector of the test frames for an RFC 2544-based benchmarking test. The logical customer edge (CE)-facing interface and **inet** family are configured on Router A. Router A is not part of a pseudowire and therefore, a Layer 3 family configuration is required on it. Router A, which is a customer edge device CE1 is connected to Router B (ACX), which functions as a provider edge device PE1 over an Ethernet pseudowire in the UNI direction with EtherType or Layer 2 Ethernet payload. The logical interface, family, and UNI direction are configured on Router B. Router B or PE1 is connected over an Ethernet pseudowire in the NNI direction to a provider edge device at the remote site, PE2. The link between CE1 and PE1 is an Ethernet Layer 2 network and it can be configured with any EtherType value. The link between PE1 and PE2 is an Ethernet line (E-Line) or an Ethernet Private Line (EPL) that has Layer 2 payload and Layer 3 transport sent over it. Router B or PE1 functions as an initiator and terminator of the test frames that are sent to Router A and reflected back from it.

This benchmarking test is used to compute the performance attributes in the user-to-network interface (UNI) direction of an Ethernet pseudowire service between Router A and Router B. Data traffic arriving from a network-to-network interface (NNI) toward the customer edge is ignored while the test is in progress. Packets from the CE are not sent toward the NNI because all packets are assumed to be test probes.

Figure 52 on page 679 shows the sample topology to perform an RFC 2544 test for the UNI direction of an Ethernet pseudowire service.

Figure 52: RFC 2544-Based Benchmarking Test for UNI Direction of an Ethernet Pseudowire



Configuration

IN THIS SECTION

In this example, you configure the benchmarking test for the UNI direction of an Ethernet pseudowire service that is enabled between two routers to detect and analyze the performance of the interconnecting routers.

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:

Configuring Benchmarking Test Parameters on Router A

```
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 0 vlan-id 101
set interfaces ge-0/0/0 unit 0 family inet address 192.0.2.1/24
set services rpm rfc2544-benchmarking profiles test-profile throughput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile throughput packet-size 64
set services rpm rfc2544-benchmarking profiles test-profile throughput test-duration 20m
set services rpm rfc2544-benchmarking profiles test-profile throughput bandwidth-kbps 500
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/0.1
set services rpm rfc2544-benchmarking tests test-name test1 test-profile throughput
set services rpm rfc2544-benchmarking tests test-name test1 mode initiate,terminate
set services rpm rfc2544-benchmarking tests test-name test1 family inet
set services rpm rfc2544-benchmarking tests test-name test1 dest-address 192.0.2.2
set services rpm rfc2544-benchmarking tests test-name test1 udp-port 4001
```

Configuring Benchmarking Test Parameters on Router B

```
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4 unit 0 encapsulation vlan-ccc
set interfaces ge-0/0/4 unit 0 vlan-id 101
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/4.1
set services rpm rfc2544-benchmarking tests test-name test1 mode reflect
set services rpm rfc2544-benchmarking tests test-name test1 mode family ccc
set services rpm rfc2544-benchmarking tests test-name test1 direction uni
```

Configuring Benchmarking Test Parameters on Router A

Step-by-Step Procedure

The following require you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router A:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

   ```
   [edit]
   user@host# edit interfaces
   ```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/0
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/0]
user@host# set vlan-tagging
```

4. Configure a logical unit and specify the protocol family as **inet**.

```
[edit interfaces ge-0/0/0]
user@host# edit unit 0 family inet
```

5. Specify the address for the logical interface.

```
[edit interfaces ge-0/0/0 unit 0 family inet]
user@host# set address 192.0.2.1/24
```

6. Configure the VLAN ID on the logical interface as 101.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
```

```
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for a test profile—for example, throughput.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile throughput
```

12. Configure the type of test to be performed as throughput.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type throughput
```

13. Specify the size of the test packet as 64 bytes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type packet-size 64
```

14. Specify the period for which the test is to be performed in hours, minutes, or seconds by specifying a number followed by the letter h (for hours), m (for minutes), or s (for seconds). In this example, you configure the period as 20 minutes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type test-duration 20m
```

15. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type bandwidth-kbps 500
```

16. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# up
```

17. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# up
```

18. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

19. Specify the name of the test profile—for example, throughput—to be associated with a particular test name.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile throughput
```

20. Specify the logical interface, ge-0/0/0.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/0.1
```

21. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode initiate-and-terminate
```

22. Configure the address type family, **inet**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

23. Configure the destination IPv4 address for the test packets as 192.0.2.2.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dest-address 192.0.2.2
```

24. Specify the UDP port of the destination to be used in the UDP header for the generated frames as 4001.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set udp-port 4001
```

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following require you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router B:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/4
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/4]
user@host# set vlan-tagging
```

4. Configure a logical unit for the interface.

```
[edit interfaces ge-0/0/4]
user@host# edit unit 0
```

5. Specify the encapsulation for Ethernet VLAN circuits.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set encapsulation vlan-ccc
```

6. Configure the VLAN ID as 101 on the logical interface.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

12. Specify the logical interface on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/4.1
```

13. Specify **reflect** as the test mode for the packets that are sent during the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

14. Configure the address type family, **ccc**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

15. Specify the direction of the interface on which the test must be run, which is UNI in this example.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction uni
```

16. Start the benchmarking test on the reflector.

```
user@host> test services rpm rfc2544-benchmarking test test1 start
```

After the test is successfully completed at the initiator, you can stop the test at the reflector by entering the **test services rpm rfc2544-benchmarking test test1 stop** command.

Results

In configuration mode, confirm your configuration on Router A and Router B by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Benchmarking Test Parameters on Router A:

```
[edit interfaces]
  ge-0/0/0 {
      vlan-tagging;
      unit 0 {
          vlan-id 101;
```

```
        family inet {
            address 192.0.2.1/24;
        }
    }
}

[edit services rpm]
rfc2544-benchmarking {
    profiles {
        test-profile throughput {
            test-type throughput
            packet-size 64;
            test-duration 20m;
            bandwidth-kbps 500;
        }
    }

    tests {
        test-name test1 {
            interface ge-0/0/0.1;
            test-profile throughput;
            mode initiate,terminate;
            family inet;
            dest-address 192.0.2.2
            udp-port 4001;
        }
    }
}
```

Benchmarking Test Parameters on Router B:

```
[edit interfaces]
  ge-0/0/4 {
      vlan-tagging;
      unit 0 {
          encapsulation vlan-ccc;
          vlan-id 101;
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
```

```
    tests {
        test-name test1 {
            interface ge-0/0/4.1;
            mode reflect;
            family ccc;
            direction uni;
        }
    }
}
```

After you have configured the device, enter the **commit** command in configuration mode.

Verifying the Results of the Benchmarking Test for UNI Direction of an Ethernet Pseudowire Service

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between Router A and Router B.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between Router A and Router B.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

Configuring an RFC 2544-Based Benchmarking Test | **659**

Example: Configuring an RFC 2544-Based Benchmarking Test on an MX104 Router for NNI Direction of Ethernet Pseudowires

IN THIS SECTION

This example shows how to configure the benchmarking test for a network-to-network interface (NNI) direction of an Ethernet pseudowire service.

Requirements

NOTE: MX Series routers support only the reflector function in RFC2544-based benchmarking tests. This example uses the MX104 3D Universal Edge Router as the reflector. You can also configure benchmarking tests on MX80 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-gigabit Ethernet MPC from Junos OS Release 16.1 or later. To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 Series routers, see "Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers" on page 666.

This example uses the following hardware and software components:

- An MX104 (reflector)
- An ACX Series router (initiator)
- Junos OS Release 13.3 or later

Overview

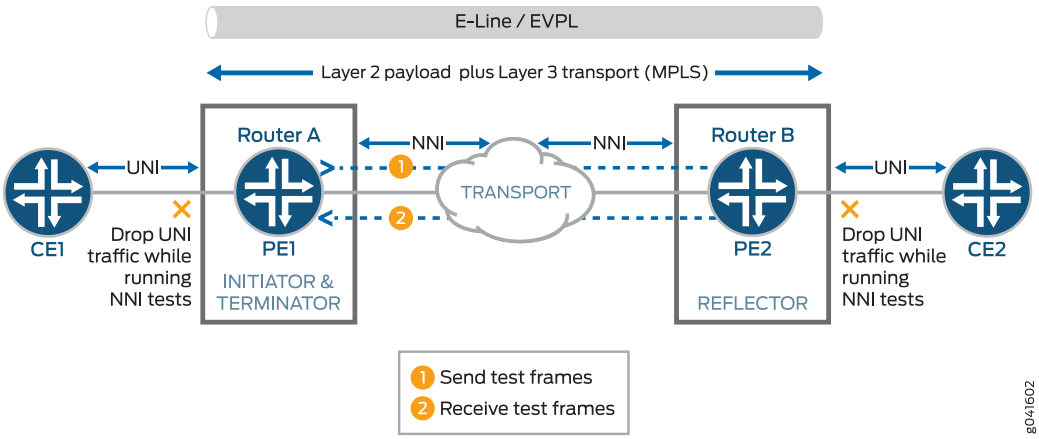
Consider a sample topology in which a router, Router A (ACX), functions as an initiator and terminator of the test frames for an RFC 2544-based benchmarking test. Router A operates as a provider edge devicePE1, which is connected to a customer edge device CE1 on one side and over an Ethernet pseudowire to another router Router B (MX104), which functions as a reflector to reflect back the test frames it receives from Router A. Router B operates as a provider edge device, PE2, which is the remote router located at the other side of the service provider core. The UNI direction of CE1 is connected to the NNI direction of PE1. An MPLS tunnel connects PE1 and PE2 over the Ethernet pseudowire or the Ethernet line (E-Line).

> **NOTE:** When pseudowire reflection is enabled on an interface, the router does not block the ingress or egress traffic through the test interface. To block other data traffic, you must explicitly configure firewall filters.

This benchmarking test is used to compute the performance attributes in the network-to-network interface (NNI) direction of an Ethernet pseudowire service between Router A and Router B. The logical interface under test on Router A is the CE1 interface with UNI as the direction, and the logical interface under test on Router B is the CE2 interface with NNI as the direction. Data traffic arriving from UNI toward NNI is ignored while the test is in progress. Packets from NNI are not sent toward the customer edge because all packets are assumed to be test frames. The family and NNI direction are configured on routers A and B.

Figure 53 on page 691 shows the sample topology to perform an RFC 2544 test for the NNI direction of an Ethernet pseudowire service.

Figure 53: RFC 2544-Based Benchmarking Test for NNI Direction of an Ethernet Pseudowire



Configuration

IN THIS SECTION

In this example, you configure the benchmarking test for the NNI direction of an Ethernet pseudowire service that is enabled between two routersto detect and analyze the performance of the interconnecting routers.

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:

Configuring Benchmarking Test Parameters on Router A

```
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 0 encapsulation vlan-ccc
```

```
set interfaces ge-0/0/0 unit 0 vlan-id 101
set services rpm rfc2544-benchmarking profiles test-profile throughput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile throughput packet-size 64
set services rpm rfc2544-benchmarking profiles test-profile throughput test-duration 20
set services rpm rfc2544-benchmarking profiles test-profile throughput bandwidth-kbps 500
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/0.1
set services rpm rfc2544-benchmarking tests test-name test1 test-profile throughput
set services rpm rfc2544-benchmarking tests test-name test1 mode initiate,terminate
set services rpm rfc2544-benchmarking tests test-name test1 family ccc
set services rpm rfc2544-benchmarking tests test-name test1 direction egress
```

Configuring Benchmarking Test Parameters on Router B

```
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4 unit 0 encapsulation vlan-ccc
set interfaces ge-0/0/4 unit 0 vlan-id 101
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/4.1
set services rpm rfc2544-benchmarking tests test-name test1 mode reflect
set services rpm rfc2544-benchmarking tests test-name test1 mode family ccc
set services rpm rfc2544-benchmarking tests test-name test1 direction ingress
```

Configuring Benchmarking Test Parameters on Router

Step-by-Step Procedure

The following require you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router A:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

   ```
   [edit]
   user@host# edit interfaces
   ```

2. Configure the interface on which the test must be run.

   ```
   [edit interfaces]
   user@host# edit ge-0/0/0
   ```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/0]
user@host# set vlan-tagging
```

4. Configure a logical unit for the interface.

```
[edit interfaces ge-0/0/0]
user@host# edit unit 0
```

5. Specify the encapsulation for Ethernet VLAN circuits.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# set encapsulation vlan-ccc
```

6. Configure the VLAN ID on the logical interface.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for a test profile—for example, throughput.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile throughput
```

12. Configure the type of test to be performed as throughput.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type throughput
```

13. Specify the size of the test packet as 64 bytes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type packet-size 64
```

14. Specify the period—for example, 20 minutes—for which the test is to be performed in hours, minutes, or seconds by specifying a number followed by the letter h (for hours), m (for minutes), or s (for seconds).

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type test-duration 20m
```

15. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type bandwidth-kbps 500
```

16. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# up
```

17. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# up
```

18. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

19. Specify the name of the test profile—for example, throughput—to be associated with a particular test name.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile throughput
```

20. Specify the logical interface, ge-0/0/0.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/0.1
```

21. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode initiate-and-terminate
```

22. Configure the address type family, **ccc**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

23. Specify the direction of the interface on which the test must be run, which is egress in this example.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction egress
```

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following require you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router B:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/4
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/4]
user@host# set vlan-tagging
```

4. Configure a logical unit for the interface.

```
[edit interfaces ge-0/0/4]
user@host# edit unit 0
```

5. Specify the encapsulation for Ethernet VLAN circuits.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set encapsulation vlan-ccc
```

6. Configure the VLAN ID on the logical interface.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

12. Specify the logical interface, ge-0/0/4.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/4.1
```

> **NOTE:** When pseudowire reflection is enabled on an interface, the router does not block the ingress or egress traffic through the test interface. To block other data traffic, you must explicitly configure firewall filters.

13. Specify **reflect** as the test mode for the packets that are sent during the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

14. Configure the address type family, **ccc**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

15. Specify the direction of the interface on which the test must be run, which is ingress in this example.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction ingress
```

16. Start the benchmarking test on the reflector.

```
user@host> test services rpm rfc2544-benchmarking test test1 start
```

After the test is successfully completed at the initiator, you can stop the test at the reflector by entering the **test services rpm rfc2544-benchmarking test test1 stop** command.

Results

In configuration mode, confirm your configuration on Router A and Router B by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Benchmarking Test Parameters on Router A:

```
[edit interfaces]
  ge-0/0/0 {
      vlan-tagging;
      unit 0 {
          encapsulation vlan-ccc;
          vlan-id 101;
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      profiles {
```

```
        test-profile throughput {
            test-type throughput
            packet-size 64;
            test-duration 20m;
            bandwidth-kbps 500;
        }
    }

    tests {
        test-name test1 {
            interface ge-0/0/0.1;
            test-profile throughput;
            mode initiate,terminate;
            family ccc;
            direction egress;
        }
    }
}
```

Benchmarking Test Parameters on Router B:

```
[edit interfaces]
  ge-0/0/4 {
      vlan-tagging;
      unit 0 {
          encapsulation vlan-ccc;
          vlan-id 101;
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
      tests {
          test-name test1 {
              interface ge-0/0/4.1;
              mode reflect;
              family ccc;
              direction egress;
          }
```

```
    }
}
```

After you have configured the device, enter the **commit** command in configuration mode.

Verifying the Results of the Benchmarking Test for NNI Direction of an Ethernet Pseudowire Service

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between Router A and Router B.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between Router A and Router B.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

Configuring an RFC 2544-Based Benchmarking Test | **659**

Example: Configuring RFC2544-Based Benchmarking Tests on an MX104 Router for Layer 2 E-LAN Services in Bridge Domains

IN THIS SECTION

This example shows how to configure benchmarking tests for the Layer 2 E-LAN services in bridge domains. The example covers the four basic tests: throughput, frame-loss, back-to-back, and latency.

Requirements

NOTE: MX Series routers support only the reflector function in RFC2544-based benchmarking tests. This example uses the MX104 3D Universal Edge Router as the reflector. You can also configure benchmarking tests on MX80 Series routers and MX240, MX480, and MX960 Series routers with MPC1, MPC2, and 16-port 10-gigabit Ethernet MPC from Junos OS Release 16.1 or later. To configure RFC2544-based benchmarking tests on MX240, MX480, MX960 Series routers, see "Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers" on page 666.

This example uses the following hardware and software components:

- An MX104 (reflector)
- An ACX Series router (initiator)
- Junos OS Release 14.2 or later for MX Series routers

Overview

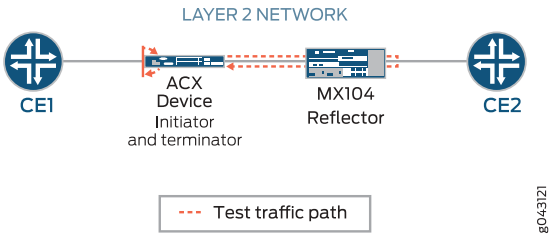
Consider a sample topology in which an ACX Series router functions as an initiator and terminator of the test frames for an RFC2544-based benchmarking test. ACX Series router is connected to a customer edge device CE1, on one side and is connected over a Layer 2 network to an MX104 Series router. The MX104

Series router functions as a reflector to reflect the test frames it receives from the ACX Series initiator back to the initiator. The MX04 Series router is also connected to a customer edge device CE2.

> **NOTE:** When Layer 2 reflection is enabled on an interface, filters are configured internally to block the ingress and egress traffic except test traffic through the test interface.

Figure 54 on page 702 shows the sample topology to perform all four RFC2544-based benchmarking tests (throughput, back-to-back frames, latency, and frame-loss) for the UNI direction on a Layer 2 bridge network.

Figure 54: Layer 2 Reflection Simple Topology



On the ACX Series router, ge-1/2/1.0 is the Layer 2 NNI interface and ge-1/1/3.0 is the Layer 2 UNI interface. On the MX104 Series router, ge-1/1/6.0 is the Layer 2 NNI interface and ge-1/1/5.0 is the Layer 2 UNI interface. The benchmarking tests are used to compute the performance attributes for an E-LAN service on a bridge domain.

> **NOTE:** Test packets can be identified using the destination MAC address, source MAC address, and test interface. Both tagged and untagged interfaces are supported. For tagged interfaces, the test interface is the VLAN sub interface. For untagged interfaces, the physical port represents the test interface. Traffic through other VLAN sub interfaces, present in the same physical port, is not affected when you configure the benchmarking test on one of the sub interfaces.

Configuration

IN THIS SECTION

In this example, you configure the benchmarking tests for the UNI direction for an E-LAN service on a Layer 2 bridge domain that is enabled between two routers to detect and analyze the performance of the interconnected routers. In this example, we start by configuring the ACX Series router. On the ACX Series router, you first configure each test by specifying the test profile, the test attributes, and then define the test by associating the test with the test profile with the relevant attributes. You can then configure the interface. On the MX104 Series router, you perform the same steps. However, a few attributes such as the outer VLAN ID, source UDP port, destination UDP port, the duration of each iteration, and their values are only applicable to the initiator or the ACX Series router.

> **NOTE:** When you configure the Layer 2 reflection, you can specify the service type under test as ELINE if you want to simulate an ELINE service using bridge encapsulation.

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:

Configuring Benchmarking Test Parameters on the ACX Series Router

```
set services rpm rfc2544-benchmarking profiles test-profile tput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile tput packet-size 128
set services rpm rfc2544-benchmarking profiles test-profile tput bandwidth-kbps 900000
set services rpm rfc2544-benchmarking profiles test-profile b2bt test-type back-back-frames
set services rpm rfc2544-benchmarking profiles test-profile b2bt packet-size 512
set services rpm rfc2544-benchmarking profiles test-profile b2bt bandwidth-kbps 950000
set services rpm rfc2544-benchmarking profiles test-profile lty test-type latency
set services rpm rfc2544-benchmarking profiles test-profile lty packet-size 512
```

```
set services rpm rfc2544-benchmarking profiles test-profile lty bandwidth-kbps 1000000
set services rpm rfc2544-benchmarking profiles test-profile frloss test-type frame-loss
set services rpm rfc2544-benchmarking profiles test-profile frloss packet-size 1600
set services rpm rfc2544-benchmarking profiles test-profile frloss bandwidth-kbps 1000000
set services rpm rfc2544-benchmarking tests test-name tput-test test-profile tput
set services rpm rfc2544-benchmarking tests test-name tput-test source-mac-address 00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name tput-test destination-mac-address
  00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name tput-test ovlan-id 400
set services rpm rfc2544-benchmarking tests test-name tput-test service-type elan
set services rpm rfc2544-benchmarking tests test-name tput-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name tput-test family bridge
set services rpm rfc2544-benchmarking tests test-name tput-test direction egress
set services rpm rfc2544-benchmarking tests test-name tput-test source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name tput-test destination-udp-port 200
set services rpm rfc2544-benchmarking tests test-name tput-test test-iterator-duration 20
set services rpm rfc2544-benchmarking tests test-name tput-test test-interface ge-1/1/3.0
set services rpm rfc2544-benchmarking tests test-name b2b-test test-profile b2bt
set services rpm rfc2544-benchmarking tests test-name b2b-test source-mac-address 00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name b2b-test destination-mac-address
  00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name b2b-test ovlan-id 400
set services rpm rfc2544-benchmarking tests test-name b2b-test service-type elan
set services rpm rfc2544-benchmarking tests test-name b2b-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name b2b-test family bridge
set services rpm rfc2544-benchmarking tests test-name b2b-test direction egress
set services rpm rfc2544-benchmarking tests test-name b2b-test test-iterator-duration 20
set services rpm rfc2544-benchmarking tests test-name b2b-test test-interface ge-1/1/3.0
set services rpm rfc2544-benchmarking tests test-name lty-test test-profile lty
set services rpm rfc2544-benchmarking tests test-name lty-test source-mac-address 00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name lty-test destination-mac-address
  00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name lty-test ovlan-id 400
set services rpm rfc2544-benchmarking tests test-name lty-test service-type elan
set services rpm rfc2544-benchmarking tests test-name lty-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name lty-test family bridge
set services rpm rfc2544-benchmarking tests test-name lty-test direction egress
set services rpm rfc2544-benchmarking tests test-name lty-test source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name lty-test destination-udp-port 200
set services rpm rfc2544-benchmarking tests test-name lty-test test-iterator-duration 20
set services rpm rfc2544-benchmarking tests test-name lty-test test-interface ge-1/1/3.0
set services rpm rfc2544-benchmarking tests test-name frloss-test test-profile frloss
```

```
set services rpm rfc2544-benchmarking tests test-name frloss-test source-mac-address
   00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name frloss-test destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name frloss-test ovlan-id 400
set services rpm rfc2544-benchmarking tests test-name frloss-test service-type elan
set services rpm rfc2544-benchmarking tests test-name frloss-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name frloss-test family bridge
set services rpm rfc2544-benchmarking tests test-name frloss-test direction egress
set services rpm rfc2544-benchmarking tests test-name frloss-test source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name frloss-test destination-udp-port 200
set services rpm rfc2544-benchmarking tests test-name frloss-test test-iterator-duration 20
set services rpm rfc2544-benchmarking tests test-name frloss-test test-interface ge-1/1/3.0
set interfaces ge-1/2/1 flexible-vlan-tagging
set interfaces ge-1/2/1 mtu 9192
set interfaces ge-1/2/1 encapsulation flexible-ethernet-services
set interfaces ge-1/2/1 unit 0 encapsulation vlan-bridge
set interfaces ge-1/2/1 unit 0 vlan-id 400
set interfaces ge-1/1/3 flexible-vlan-tagging
set interfaces ge-1/1/3 mtu 9192
set interfaces ge-1/1/3 encapsulation flexible-ethernet-services
set interfaces ge-1/1/3 unit 0 encapsulation vlan-bridge
set interfaces ge-1/1/3 unit 0 vlan-id 400
set bridge-domains bd1 vlan-id 600
set bridge-domains bd1 interface ge-1/2/1.0
set bridge-domains bd1 interface ge-1/1/3.0
```

Configuring Benchmarking Test Parameters on the MX104 Router

```
set services rpm rfc2544-benchmarking tests test-name l2b-reflector source-mac-address
   00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name l2b-reflector destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name l2b-reflector service-type elan
set services rpm rfc2544-benchmarking tests test-name l2b-reflector mode reflect
set services rpm rfc2544-benchmarking tests test-name l2b-reflector family bridge
set services rpm rfc2544-benchmarking tests test-name l2b-reflector direction egress
set services rpm rfc2544-benchmarking tests test-name l2b-reflector test-interface ge-1/1/5.0
set interfaces ge-1/1/6 flexible-vlan-tagging
set interfaces ge-1/1/6 mtu 9192
```

```
set interfaces ge-1/1/6 encapsulation flexible-ethernet-services
set interfaces ge-1/1/6 unit 0 encapsulation vlan-bridge
set interfaces ge-1/1/6 unit 0 vlan-id 400
set interfaces ge-1/1/5 flexible-vlan-tagging
set interfaces ge-1/1/5 mtu 9192
set interfaces ge-1/1/5 encapsulation flexible-ethernet-services
set interfaces ge-1/1/5 unit 0 encapsulation vlan-bridge
set interfaces ge-1/1/5 unit 0 vlan-id 400
set bridge-domains bd1 domain-type bridge
set bridge-domains bd1 vlan-id 500
set bridge-domains bd1 interface ge-1/1/6.0
set bridge-domains bd1 interface ge-1/1/5.0
```

Configuring Throughput Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the throughput test and reference the test-profile in a unique test-name. The test-name defines the parameters for the throughput test to be performed on the ACX Series router.

To configure the throughput test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

   ```
   [edit]
   user@host# edit services rpm rfc2544-benchmarking
   ```

2. Define a name for the first test profile—for example, tput for the throughput test profile.

   ```
   [edit services rpm rfc2544-benchmarking]
   user@host# edit profiles test-profile tput
   ```

3. Configure the type of test to be performed as throughput, specify the packet size as 128 bytes, and define the theoretical maximum bandwidth for the test in kilobits per second (Kbps), with a value from 1 Kbps through 1,000,000 Kbps.

   ```
   [edit services rpm rfc2544-benchmarking profiles test-profile tput]
   user@host# set test-type throughput packet-size 128 bandwidth-kbps 900000
   ```

4. Enter the **up** command twice to go to the [**edit services rpm rfc2544-benchmarking**] level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile tput ]
user@host# up
user@host# up
```

5. Define a name for the throughput test—for example, tput-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name tput-test
```

6. Specify the name of the test profile, tput, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set test-profile tput
```

7. Configure the source and destination MAC address for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test to be E-LAN.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set ovlan-id 400 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress. Also, specify the source and destination UDP port to be used in the UDP headers of the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
```

```
user@host# set family bridge direction egress source-udp-port 200 destination-udp-port 200
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds, and specify the logical interface, ge-0/2/1.0, on which the RFC2544-benchmarking tests are run.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set test-iterator-duration 20 test-interface ge-1/1/3.0
```

Configuring Back-to-Back Frames Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the back to back frames test and reference the test-profile in a unique test-name. The test-name defines the parameters for the back to back frames test to be performed on the ACX Series router.

To configure the back-to-back frames test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the back-to-back test profile—for example, b2bt.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile b2bt
```

3. Configure the type of test to be performed as back-to-back frames, specify the packet size as 128 bytes, and define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile b2bt]
user@host# set test-type back-toback-frames packet-size 4444 bandwidth-kbps 950000
```

4. Enter the **up** command twice to go to the [**edit services rpm rfc2544-benchmarking**] level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile b2bt ]
```

```
user@host# up
user@host# up
```

5. Define a name for the back-to-back frames test—for example, b2bt-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name b2bt-test
```

6. Specify the name of the test profile, b2bt, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set test-profile b2bt
```

7. Configure the source and destination MAC address for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set ovlan-id 400 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set family bridge direction egress
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds. Also, specify the logical interface, ge-0/2/1.0, on which the RFC2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set test-iterator-duration 20 test-interface ge-1/1/3.0
```

Configuring Latency Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the latency test and reference the test-profile in a unique test-name. The test-name defines the parameters for the latency test to be performed on the ACX Series router.

To configure the latency test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

   ```
   [edit]
   user@host# edit services rpm rfc2544-benchmarking
   ```

2. Define a name for the latency test profile—for example, lty.

   ```
   [edit services rpm rfc2544-benchmarking]
   user@host# edit profiles test-profile lty
   ```

3. Configure the type of test to be performed as latency, specify the packet size of the test packet, and define the maximum bandwidth for the test in kilobits per second, with a value form 1 Kbps through 1,000,000 Kbps.

   ```
   [edit services rpm rfc2544-benchmarking profiles]
   user@host# set test-profile lty test-type latency packet-size 512 bandwidth-kbps 1000000
   ```

4. Enter the **up** command twice to go to the previous level in the configuration hierarchy.

   ```
   [edit services rpm rfc2544-benchmarking profiles test-profile lty]
   user@host# up
   user@host# up
   ```

5. Define a name for the latency test—for example, lty-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name lty-test
```

6. Specify the name of the test profile, lty, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set test-profile lty
```

7. Configure the source and destination MAC address for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set ovlan-id 400 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress. Also, specify the source and destination UDP port to be used in the UDP headers of the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set family bridge direction egress source-udp-port 200 destination-udp-port 200
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds. Also, specify the logical interface, ge-0/2/1.0, on which the RFC2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set test-iterator-duration 20 test-interface ge-1/1/3.0
```

Configuring Frame Loss Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the frame loss test and reference the test-profile in a unique test-name. The test-name defines the parameters for the frame loss test to be performed on the ACX Series router.

To configure the frame loss test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the frame loss test profile—for example, frloss.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile frloss
```

3. Configure the type of test performed as frame loss, specify the packet size of the test packet, and define the maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# set test-profile frloss test-type frame-loss packet-size 1600 bandwidth-kbps 1000000
```

4. Enter the **up** command to go to the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles ]
user@host# up
```

5. Define a name for the frame loss test—for example, frloss-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name frloss-test
```

6. Specify the name of the test profile, frloss, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set test-profile frloss
```

7. Configure the source and destination MAC address for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set ovlan-id 400 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress. Also, specify the source and destination UDP port to be used in the UDP headers of the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set family bridge direction egress source-udp-port 200 destination-udp-port 200
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds. Also, specify the logical interface, ge-0/2/1.0, on which the RFC2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set test-iterator-duration 20 test-interface ge-1/1/3.0
```

12. Enter the **exit** command to go to the [edit] hierarchy level.

```
[edit services rpm rfc2544-benchmarking tests test-name test4 ]
user@host# exit
```

Configuring Other Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the interface and bridge domain on the ACX Series router:

1. Configure the Layer 2 NNI interface on which the tests must be run from the [**edit**] hierarchy level.

```
[edit]
user@host# edit interfaces ge-1/2/1
```

2. Configure flexible VLAN tagging for the transmission of untagged frames or 802.1Q single-tagged and dual-tagged frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-1/2/1]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation flexible-ethernet-services
```

3. Configure a logical unit for the interface, specify the encapsulation, and configure the VLAN ID on the logical interfaces.

```
[edit interfaces ge-1/2/1]
user@host# set unit 0 encapsulation vlan-bridge vlan-id 400
```

4. Configure the Layer 2 UNI interface.

```
[edit]
user@host# edit interfaces ge-1/1/3
```

5. Configure flexible VLAN tagging for transmission of non-tagged frames or 802.1Q single-tag and dual-tag frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-1/1/3]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation flexible-ethernet-services
```

6. Configure a logical unit for the interface and specify the encapsulation and configure the VLAN ID on the logical interfaces.

```
[edit interfaces ge-1/1/3]
user@host# set unit 0 encapsulation vlan-bridge vlan-id 400
```

7. Configure the bridge domain, bd1, and specify the VLAN ID associated with the bridge domain and the associated interfaces from the [**edit**] hierarchy level.

```
[edit ]
user@host# set bridge-domains bd1 vlan-id 600 interface ge-1/2/1.0
user@host# set bridge-domains bd1 vlan-id 600 intreface ge-1/1/3.0
```

Configuring Benchmarking Test Parameters on the MX104 Router

Step-by-Step Procedure

The following configuration requires you to configure a unique test-name for the benchmarking test on the MX104 Series router. The test-name defines the parameters for the benchmarking test to be performed. Because the test interface and test MAC addresses are the same, you can create a single test configuration at the reflector (MX104).

To configure the benchmarking test parameters on the MX104 Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the test—for example, l2b-reflector. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name l2b-reflector
```

3. Specify the source and destination MAC addresses of the test packet.

```
[edit services rpm rfc2544-benchmarking test-name l2b-reflector]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

4. Specify the service type under test and the mode which is reflect, at the reflector.

```
[edit services rpm rfc2544-benchmarking test-name l2b-reflector]
user@host# set service-type elan
```

5. Specify the mode which is reflect at the reflector.

```
[edit services rpm rfc2544-benchmarking test-name l2b-reflector]
user@host# set mode reflect
```

6. Configure the family type, **bridge** and specify the direction, egress, for the benchmarking test. Also, specify the logical interface, ge-1/1/5.0, on which the RFC2544-based benchmarking test is being run.

```
[edit services rpm rfc2544-benchmarking tests test-name l2b-reflector]
user@host# set family bridge direction egress test-interface ge-1/1/5.0
```

Configuring Other Benchmarking Test Parameters on the MX104 Router

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the interface and bridge domain on the MX104 Series router:

1. Configure the Layer 2 NNI interface on which the tests must be run.

```
[edit]
user@host# edit interfaces ge-1/1/6
```

2. Configure flexible VLAN tagging for transmission of untagged frames or 802.1Q single-tagged and dual-tagged frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-1/1/6]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation flexible-ethernet-services
```

3. Configure a logical unit for the interface, specify the encapsulation, and configure the VLAN ID on the logical interface.

```
[edit interfaces ge-1/1/6]
user@host# set unit 0 encapsulation vlan-bridge vlan-id 400
```

4. Configure the Layer 2 NNI interface.

```
[edit]
user@host# edit interfaces ge-1/1/5
```

5. Configure flexible VLAN tagging for transmission of untagged frames or 802.1Q single-tagged and dual-tagged frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-1/1/5]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation flexible-ethernet-services
```

6. Configure a logical unit for the interface, specify the encapsulation, and configure the VLAN ID on the logical interfaces.

```
[edit interfaces ge-1/1/5]
user@host# set unit 0 encapsulation vlan-bridge vlan-id 400
```

7. Configure the bridge domain, bd1, and specify the VLAN ID associated with the bridge domain, and the associated interfaces from the [**edit**] hierarchy level.

```
[edit ]
user@host# set bridge-domains bd1 vlan-id 500 interface ge-1/1/6.0
user@host# set bridge-domains bd1 vlan-id 500 intreface ge-1/1/5.0
```

8. Start the benchmarking test on the reflector.

```
user@host> test services rpm rfc2544-benchamrking test l2b-reflector start
```

After the test is successfully completed at the initiator, you can stop the test at the reflector by entering the **test services rpm rfc2544-benchmarking test l2b-reflector stop** command.

Results

In configuration mode, confirm your configuration on the ACX Series router and the MX104 Series router by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Benchmarking Test Parameters on the ACX Series router :

```
[edit interfaces]
  ge-1/2/1 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation flexible-ethernet-services;
      unit 0 {
encapsulation vlan-bridge;
          vlan-id 400;
        }
  }
ge-1/1/3 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation flexible-ethernet-services;
      unit 0 {
encapsulation vlan-bridge;
          vlan-id 400;
             }
  }

[edit bridge-domains]
 bd1 {
    vlan-id 600;
    interface ge-1/2/1.0;
    interface ge-1/1/3.0;
}

  [edit services rpm]
  rfc2544-benchmarking {
      profiles {
          test-profile tput {
              test-type throughput
              packet-size 128;
              bandwidth-kbps 900000;
          }
test-profile b2bt {
              test-type back-back-frames
              packet-size 512;
             bandwidth-kbps 950000;
          }
test-profile lty {
              test-type latency
              packet-size 512;
              bandwidth-kbps 100000;
```

```
          }
test-profile frloss {
              test-type frameloss
              packet-size 1600;
              bandwidth-kbps 1000000;
          }
              }

      tests {
          test-name tput-test {
              interface ge-1/1/3.0;
              test-profile tput;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 400;
              service-type elan;
              family bridge;
  direction egress;
  source-udp-port 200;
  destination-udp-port 200;
  test-iterator-duration 20;
          }
test-name b2b-test {
              interface ge-1/1/3.0;
              test-profile b2bt;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 400;
              service-type elan;
              family bridge;
  direction egress;
  test-iterator-duration 20;
          }
test-name lty-test {
              interface ge-1/1/3.0;
              test-profile lty;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 400;
              service-type elan;
              family bridge;
```

```
  direction egress;
  source-udp-port 200;
  destination-udp-port 200;
  test-iterator-duration 20;
          }
test-name frloss-test {
              interface ge-1/1/3.0;
              test-profile frloss;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 400;
              service-type elan;
              family bridge;
  direction egress;
  source-udp-port 200;
  destination-udp-port 200;
  test-iterator-duration 20;
          }
      }
  }
```

Benchmarking Test Parameters on the MX104 Series router:

```
[edit interfaces]
  ge-1/1/6 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation flexible-ethernet-services;
      unit 0 {
encapsulation vlan-bridge;
          vlan-id 400;
        }
  }
ge-1/1/5 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation flexible-ethernet-services;
      unit 0 {
encapsulation vlan-bridge;
          vlan-id 400;
        }
  }
```

```
    }
}
[edit bridge-domains]
 bd1 {
    vlan-id 500;
    interface ge-1/1/6.0;
    interface ge-1/1/5.0;

}

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
      tests {
          test-name l2b-reflector {
              interface ge-1/1/5.0;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  family bridge;
  mode reflect;
              service-type elan;
              family bridge;
  direction egress;
            }
      }
  }
```

Verifying the Results of the Benchmarking Tests for Layer 2 Services (E-LAN) in Bridge Domains

IN THIS SECTION

Examine the results of the benchmarking tests that are performed on the configured service between the ACX Series router and the MX104 Series router. Start the test on the reflector first and then start the test on the initiator.

Verifying the Throughput Benchmarking Test Results

Purpose

Verify that the necessary and statistical values are displayed for the benchmarking tests that are run on the configured service between the ACX Series router and the MX104 Series router.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the ACX Series router.

```
user@host> show services rpm rfc2544-benchmarking test-id 1 detail
```

```
Test information :
      Test id: 1, Test name: tput_test, Test type: Throughput
      Test mode: Initiate-and-Terminate
      Test packet size: 128
      Test state: TEST_STATE_COMPLETED
      Status: Test-Completed
      Test start time: 2014-09-24 22:21:09 PDT
      Test finish time: 2014-09-24 22:21:33 PDT
      Counters last cleared: Never

      Test-profile Configuration:
          Test-profile name: tput
          Test packet size: 128
          Theoretical max bandwidth : 900000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 20
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/3.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Outer vlan-id: 400
          Outer vlan priority: 0
          Outer vlan cfi: 0
          Outer tag protocol id: 0x8100
```

```
          Source ipv4 address: 192.168.1.10
          Destination ipv4 address: 192.168.1.20
          Source udp port: 200
          Destination udp port: 200

  Rfc2544 throughput test information :
      Initial test load percentage : 100.00 %
      Test iteration mode : Binary
      Test iteration step : 50.00 %
      Theoretical max bandwidth : 900000 kbps

Test packet size: 128
Iteration  Internal Duration  Elapsed  --------  Throughput ---------
           Overhead  (sec)     time    Theoretical  Transmit Measured
   1         0       20        20       100.00 %    100.00 % 100.00 %

Result of the iteration runs : Throughput Test complete for packet size 128
Best iteration : 1, Best iteration (pps) : 760135
Best iteration throughput : 100.00 %

RFC2544 Throughput test results summary:
----------------------------------------

  Packet  Internal    Theoretical Transmit    Tx           Rx       Measured
   Measured
   Size   overhead    rate (pps)    pps     Packets      Packets   throughput %
 bandwidth (kbps)
   128     0           760135     760135    15202700     15202700    100.00 %
   900000
```

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the MX104 Series router.

```
user@host> show services rpm rfc2544-benchmarking test-id 1 detail
```

```
Test information :
      Test id: 1, Test name: l2b-reflector, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: TEST_STATE_RUNNING
      Status: Running
      Test start time: 2014-09-24 22:20:54 PDT
      Test finish time: TEST_RUNNING
```

```
      Counters last cleared: Never

      Test Configuration:
          Test mode: Reflect
          Duration in seconds: 864000
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/5.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Service type: Elan

Elapsed        Reflected            Reflected
 time           Packets              Bytes
     61        15202700            1945945600
```

You can also use the **show services rpm rfc2544-benchmarking (aborted-test | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC2544-based benchmarking tests for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

Verifying the Back-to-Back Benchmarking Test Results

Purpose

Verify that the necessary and statistical values are displayed for the benchmarking tests that are run on the configured service between the ACX Series router and the MX104 Series router.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the ACX Series router.

`user@host>` **show services rpm rfc2544-benchmarking test-id 4 detail**

```
Test information :
      Test id: 4, Test name: b2b-test, Test type: Back-Back-Frames
      Test mode: Initiate-and-Terminate
      Test packet size: 128 512
      Test state: TEST_STATE_COMPLETED
      Status: Test-Completed
      Test start time: 2014-09-24 22:30:16 PDT
      Test finish time: 2014-09-24 22:31:03 PDT
      Counters last cleared: Never

      Test-profile Configuration:
          Test-profile name: b2bt
          Test packet size: 128 512
          Theoretical max bandwidth : 950000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 20
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/3.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Outer vlan-id: 400
          Outer vlan priority: 0
          Outer vlan cfi: 0
          Outer tag protocol id: 0x8100
          Source ipv4 address: 192.168.1.10
          Destination ipv4 address: 192.168.1.20
          Source udp port: 4040
          Destination udp port: 4041

    Rfc2544 Back-Back test information :
      Initial burst length: 20 seconds at 950000 kbps
      Test iteration mode : Binary
      Test iteration step : 50.00 %

Test packet size: 128
```

```
Iteration   Theoretical    Transmit     Internal Duration  Elapsed
            burst length  burst length  overhead   time
             (packets)     (packets)
    1    16047280         16047280        0       20        20

Result of the iteration runs : Back-Back Test complete for packet size 128
Best iteration : 1
Measured burst (num sec)  : 20 sec
Measured burst (num pkts)  : 16047280 packets

Test packet size: 512
Iteration   Theoretical    Transmit     Internal Duration  Elapsed
            burst length  burst length  overhead   time
             (packets)     (packets)
    1    4464280          4464280         0       20        20

Result of the iteration runs : Back-Back Test complete for packet size 512
Best iteration : 1
Measured burst (num sec)  : 20 sec
Measured burst (num pkts)  : 4464280 packets

RFC2544 Back-Back test results summary:
----------------------------------------

  Packet      Measured Burst      Time
   Size      length (Packets)   (seconds)
    128           16047280        20
    512            4464280        20
```

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the MX104 Series router.

`user@host>` **show services rpm rfc2544-benchmarking test-id 4 detail**

```
Test information :
      Test id: 4, Test name: l2b-reflector, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: TEST_STATE_RUNNING
      Status: Running
      Test start time: 2014-09-24 22:30:07 PDT
      Test finish time: TEST_RUNNING
      Counters last cleared: Never
```

```
      Test Configuration:
          Test mode: Reflect
          Duration in seconds: 864000
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/5.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Service type: Elan

Elapsed        Reflected           Reflected
 time           Packets             Bytes
     58        20511560           4339763200
```

You can also use the **show services rpm rfc2544-benchmarking (aborted-test | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC2544-based benchmarking tests for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

Verifying the Frame Loss Benchmarking Test Results

Purpose

Verify that the necessary and statistical values are displayed for the benchmarking tests that are run on the configured service between the ACX Series router and the MX104 Series router.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the ACX Series router.

```
user@host> show services rpm rfc2544-benchmarking test-id 3 detail
```

```
Test information :
      Test id: 3, Test name: frloss-test, Test type: Frame-Loss
      Test mode: Initiate-and-Terminate
      Test packet size: 1600
      Test state: TEST_STATE_COMPLETED
      Status: Test-Completed
      Test start time: 2014-09-24 22:26:45 PDT
      Test finish time: 2014-09-24 22:27:55 PDT
      Counters last cleared: Never

      Test-profile Configuration:
          Test-profile name: frloss
          Test packet size: 1600
          Theoretical max bandwidth : 1000000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 20
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/3.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Outer vlan-id: 400
          Outer vlan priority: 0
          Outer vlan cfi: 0
          Outer tag protocol id: 0x8100
          Source ipv4 address: 192.168.1.10
          Destination ipv4 address: 192.168.1.20
          Source udp port: 200
          Destination udp port: 200

  Rfc2544 frame-loss test information :
      Initial test load percentage : 100.00 %
      Test iteration mode : step-down
      Test iteration step : 10 %
      Theoretical max bandwidth : 1000000 kbps
```

```
Test packet size: 1600
Iteration Internal Duration  Elapsed  --------  Throughput ---------  Frame-loss
          Overhead  (sec)     time    Theoretical  Transmit Measured    rate %
   1        0        20       20       100.00 %     100.00 % 100.00 %  0.00 %
   2        0        20       20       100.00 %     100.00 % 100.00 %  0.00 %
   3        0        20       20       100.00 %     100.00 % 100.00 %  0.00 %

Result of the iteration runs : Frame-loss test complete for packet size 1600
Percentage throughput transmitted: 100.00 %
Frame-loss rate (percent) : 0.00 %

RFC2544 Frame-loss test results summary:
----------------------------------------

  Packet  Internal  Theoretical  Transmit    Transmit       Tx             Rx
       Frame Loss
   Size   overhead  rate (pps)      pps     throughput    Packets        Packets
      rate percent
    1600   0         77160       77160      100.00 %      1543200        1543200
       0.00 %
```

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the MX104 Series router.

```
user@host> show services rpm rfc2544-benchmarking test-id 3 detail
```

```
Test information :
      Test id: 3, Test name: l2b-reflector, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: TEST_STATE_RUNNING
      Status: Running
      Test start time: 2014-09-24 22:25:36 PDT
      Test finish time: TEST_RUNNING
      Counters last cleared: Never

      Test Configuration:
          Test mode: Reflect
          Duration in seconds: 864000
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
```

```
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/5.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Service type: Elan

Elapsed        Reflected            Reflected
 time           Packets              Bytes
     95         1624361            2598977600
```

You can also use the **show services rpm rfc2544-benchmarking (aborted-test | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC2544-based benchmarking tests for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

Verifying the Latency Benchmarking Test Results

Purpose

Verify that the necessary and statistical values are displayed for the benchmarking tests that are run on the configured service between the ACX Series router and the MX104 Series router.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the ACX Series router.

`user@host>` **show services rpm rfc2544-benchmarking test-id 5 detail**

```
Test information :
      Test id: 5, Test name: lty-test, Test type: Latency
      Test mode: Initiate-and-Terminate
      Test packet size: 512
      Test state: TEST_STATE_COMPLETED
      Status: Test-Completed
      Test start time: 2014-09-24 22:33:05 PDT
      Test finish time: 2014-09-24 22:40:46 PDT
      Counters last cleared: Never
```

```
      Test-profile Configuration:
          Test-profile name: lty
          Test packet size: 512
          Theoretical max bandwidth : 1000000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 20
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/3.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Outer vlan-id: 400
          Outer vlan priority: 0
          Outer vlan cfi: 0
          Outer tag protocol id: 0x8100
          Source ipv4 address: 192.168.1.10
          Destination ipv4 address: 192.168.1.20
          Source udp port: 200
          Destination udp port: 200

  Rfc2544 latency test information :
      Theoretical max bandwidth : 1000000 kbps
      Initial test load percentage : 100.00 %
      Duration in seconds: 20
      Measurement unit for timestamp: Nanoseconds

Test packet size: 512
Iteration  Duration   Elapsed  Theoretical   Transmit      Throughput
------------------ Latency --------------------
            (sec)      time    rate (pps)       pps      percent          Minimum
     Average      Maximum     Probe
   1        20        20      234962         234962        100.00 %        44008
      45253        47424       45096
   2        20        20      234962         234962        100.00 %        44008
      45237        47456       45256
```

```
   3        20        20      234962          234962        100.00 %        43864
      45198        46976       45144
   4        20        20      234962          234962        100.00 %        43832
      45243        47088       45096
   5        20        20      234962          234962        100.00 %        44072
      45261        46976       45176
   6        20        20      234962          234962        100.00 %        43784
      45214        46864       45032
   7        20        20      234962          234962        100.00 %        44024
      45259        47216       45240
   8        20        20      234962          234962        100.00 %        44072
      45290        46864       45192
   9        20        20      234962          234962        100.00 %        43976
      45272        46792       45208
   10       20        20      234962          234962        100.00 %        44024
      45206        46976       45112
   11       20        20      234962          234962        100.00 %        44040
      45198        47088       45176
   12       20        20      234962          234962        100.00 %        44008
      45223        46976       45160
   13       20        20      234962          234962        100.00 %        44088
      45257        47408       45176
   14       20        20      234962          234962        100.00 %        43976
      45183        46832       45080
   15       20        20      234962          234962        100.00 %        44024
      45198        47088       45112
   16       20        20      234962          234962        100.00 %        43864
      45206        46912       45208
   17       20        20      234962          234962        100.00 %        44056
      45209        46960       45176
   18       20        20      234962          234962        100.00 %        44008
      45198        46912       45112
   19       20        20      234962          234962        100.00 %        43816
      45175        47248       45000
   20       20        20      234962          234962        100.00 %        43912
      45202        46992       45192

Result of the iteration runs : Latency Test complete for packet size 512
Internal overhead per packet: 0
Avg (min) Latency                  : 43972
Avg (avg) latency                  : 45224
Avg (Max) latency                  : 47052
Avg (probe) latency                : 45147
```

```
RFC2544 Latency test results summary:
---------------------------------------

  Packet Internal  Theoretical  Transmit      Tx           Rx
------------------ Latency --------------------
   Size  overhead  rate (pps)      pps     Packets       Packets      Minimum
 Average      Maximum     Probe
   512     0        234962      234962      93984800      93984800     43972
  45224        47052       45147
```

In operational mode, enter the **show services rpm rfc2544-benchmarking test-id *test-id-number* detail** command on the MX104 Series router.

`user@host>` **show services rpm rfc2544-benchmarking test-id 5 detail**

```
Test information :
      Test id: 5, Test name: l2b-reflector, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: TEST_STATE_RUNNING
      Status: Running
      Test start time: 2014-09-24 22:32:55 PDT
      Test finish time: TEST_RUNNING
      Counters last cleared: Never

      Test Configuration:
          Test mode: Reflect
          Duration in seconds: 864000
          Test finish wait duration in seconds: 1
          Test family: Bridge
          Test iterator pass threshold: 0.50 %
          Test receive failure threshold: 0.00 %
          Test transmit failure threshold: 0.50 %

      Bridge family Configuration:
          Interface : ge-1/1/5.0
          Test direction: Egress
          Source mac address: 00:00:5e:00:53:11
          Destination mac address: 00:00:5e:00:53:22
          Service type: Elan

Elapsed        Reflected           Reflected
```

```
time            Packets              Bytes
   426        84586320          43308195840
```

You can also use the **show services rpm rfc2544-benchmarking (aborted-test | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC2544-based benchmarking tests for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests for E-LAN and E-Line Services on MX Series Routers | **652**

Supported RFC2544-Based Benchmarking Statements on MX Series Routers | **657**

Example: Configuring Benchmarking Tests to Measure SLA Parameters for E-LAN Services on an MX104 Router Using VPLS

IN THIS SECTION

This example shows how to configure benchmarking tests for the E-LAN services using BGP-based VPLS. The example covers the four benchmarking tests: throughput, frame loss, back-to-back frames, and latency.

Requirements

> **NOTE:** MX Series routers support only the reflector function in RFC2544-based benchmarking tests.

This example uses the following hardware and software components:

- An MX104 3D Universal Edge Router (reflector)
- Any MX Series router
- Any ACX Series router (initiator)
- Junos OS Release 15.1 or later for MX Series routers

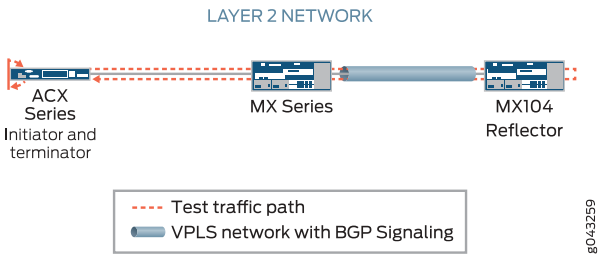
Overview

Consider a sample topology in which an ACX Series router functions as an initiator and terminator of the test frames for an RFC2544-based benchmarking test. The ACX Series router is connected to a provider edge router, PE1 (an MX Series router). The PE1 router is configured with a VPLS routing instance and is connected over a Layer 2 network to another provider edge router, PE2 (an MX104 Series router). A simple VPLS network with BGP signalling is created between routers PE1 and PE2. The MX104 Series router also functions as a reflector to reflect the test frames it receives from the ACX Series router back to the initiator.

Benchmarking tests compute the performance attributes in the user-to-network interface (UNI) direction of the Layer 2 E-LAN service between the ACX Series router and the MX104 Series router. To measure SLA parameters for E-LAN services using VPLS, configure specific benchmarking tests. In this example, all four benchmarking tests (throughput, back-to-back frames, latency, and frame-loss) are configured.

Figure 55 on page 735 shows the sample topology to perform all four RFC2544-based benchmarking tests for the UNI direction on a Layer 2 network using VPLS.

Figure 55: Layer 2 Reflection with Simple BGP-based VPLS Topology



On the ACX Series router, ge-0/2/1.0 is the Layer 2 NNI interface and ge-0/2/0.0 is the Layer 2 UNI interface. For each benchmarking test configured on the ACX Series router, specify the source MAC address as 00:00:5e:00:53:11 and 00:00:5e:00:53:22 as the destination MAC address. Also, specify the VLAN ID as 512. On the MX Series router, ge-0/3/0.0 is the Layer 2 NNI interface and ge-0/2/1.0 is the UNI interface. On the MX104 Series router, ge-0/2/5.0 is the Layer 2 NNI interface and ge-0/3/1.0 is the Layer 2 UNI interface. The benchmarking tests are used to compute the performance attributes for an E-LAN service using VPLS.

Configuration

IN THIS SECTION

In this example, you configure the benchmarking tests for the UNI direction for a Layer 2 E-LAN service using VPLS between two routers (initiator and reflector) to detect and analyze the performance of the interconnected routers. The initiator and reflector routers are not directly connected to each other. The initiator is connected to a provider edge router (PE1), which is in turn connected to the reflector. In this example, the ACX Series router is the initiator, an MX Series router is PE1, and the MX104 router is the other provider edge router (PE2) and reflector. Start by configuring the initiator. On the ACX Series router, you first configure each test by specifying the test profile and the test attributes, and then define the test by associating the test with the test profile with the relevant attributes. You can then configure the interface. On the MX Series router, configure the VPLS parameters to enable VPLS on the router. On the MX104 Series router, configure the benchmarking parameters and the VPLS parameters.

NOTE: When you configure Layer 2 reflection, you can specify the service type under test as ELINE if you want to simulate an Eline service by using bridge encapsulation.

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:

Configuring Benchmarking Test Parameters on the ACX Series Router (Initiator)

```
set services rpm rfc2544-benchmarking profiles test-profile tput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile tput packet-size 256
set services rpm rfc2544-benchmarking profiles test-profile tput bandwidth-kbps 600000
set services rpm rfc2544-benchmarking profiles test-profile b2bt test-type back-back-frames
set services rpm rfc2544-benchmarking profiles test-profile b2bt packet-size 9104
set services rpm rfc2544-benchmarking profiles test-profile b2bt bandwidth-kbps 600000
set services rpm rfc2544-benchmarking profiles test-profile lty test-type latency
set services rpm rfc2544-benchmarking profiles test-profile lty packet-size 1024
set services rpm rfc2544-benchmarking profiles test-profile lty bandwidth-kbps 6000000
set services rpm rfc2544-benchmarking profiles test-profile frloss test-type frame-loss
set services rpm rfc2544-benchmarking profiles test-profile frloss packet-size 1600
set services rpm rfc2544-benchmarking profiles test-profile frloss bandwidth-kbps 6000000
set services rpm rfc2544-benchmarking profiles test-profile frloss step-percent 5
set services rpm rfc2544-benchmarking tests test-name tput-test test-profile tput
set services rpm rfc2544-benchmarking tests test-name tput-test source-mac-address 00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name tput-test destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name tput-test ovlan-id 512
set services rpm rfc2544-benchmarking tests test-name tput-test service-type elan
set services rpm rfc2544-benchmarking tests test-name tput-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name tput-test family bridge
set services rpm rfc2544-benchmarking tests test-name tput-test direction egress
set services rpm rfc2544-benchmarking tests test-name tput-test source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name tput-test destination-udp-port 400
set services rpm rfc2544-benchmarking tests test-name tput-test test-iterator-duration 250
set services rpm rfc2544-benchmarking tests test-name tput-test test-interface ge-0/2/0.0
set services rpm rfc2544-benchmarking tests test-name b2bt-test test-profile b2bt
set services rpm rfc2544-benchmarking tests test-name b2bt-test source-mac-address 00:00:5e:00:53:11
```

```
set services rpm rfc2544-benchmarking tests test-name b2bt-test destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name b2bt-test ovlan-id 512
set services rpm rfc2544-benchmarking tests test-name b2bt-test service-type elan
set services rpm rfc2544-benchmarking tests test-name b2bt-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name b2bt-test family bridge
set services rpm rfc2544-benchmarking tests test-name b2bt-test direction egress
set services rpm rfc2544-benchmarking tests test-name b2bt--test destination-udp-port 400
set services rpm rfc2544-benchmarking tests test-name b2bt-test test-iterator-duration 10
set services rpm rfc2544-benchmarking tests test-name b2b-test test-interface ge-0/2/0.0
set services rpm rfc2544-benchmarking tests test-name lty-test test-profile lty
set services rpm rfc2544-benchmarking tests test-name lty-test source-mac-address 00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name lty-test destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name lty-test ovlan-id 512
set services rpm rfc2544-benchmarking tests test-name lty-test service-type elan
set services rpm rfc2544-benchmarking tests test-name lty-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name lty-test family bridge
set services rpm rfc2544-benchmarking tests test-name lty-test direction egress
set services rpm rfc2544-benchmarking tests test-name lty-test source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name lty-test destination-udp-port 400
set services rpm rfc2544-benchmarking tests test-name lty-test test-iterator-duration 10
set services rpm rfc2544-benchmarking tests test-name lty-test test-interface ge-0/2/0.0
set services rpm rfc2544-benchmarking tests test-name frloss-test test-profile frloss
set services rpm rfc2544-benchmarking tests test-name frloss-test source-mac-address
   00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name frloss-test destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name frloss-test ovlan-id 512
set services rpm rfc2544-benchamrking tests test-name frloss-test ovlan-priority 7
set services rpm rfc2544-benchamrking tests test-name frloss-test ovlan-cfi 1
set services rpm rfc2544-benchmarking tests test-name frloss-test service-type elan
set services rpm rfc2544-benchmarking tests test-name frloss-test mode initiate-and-terminate
set services rpm rfc2544-benchmarking tests test-name frloss-test family bridge
set services rpm rfc2544-benchmarking tests test-name frloss-test direction egress
set services rpm rfc2544-benchmarking tests test-name frloss-test source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name frloss-test destination-udp-port 400
set services rpm rfc2544-benchmarking tests test-name frloss-test test-iterator-duration 30
set services rpm rfc2544-benchmarking tests test-name frloss-test test-interface ge-0/2/0.0
set interfaces ge-0/2/0 flexible-vlan-tagging
set interfaces ge-0/2/0 mtu 9192
set interfaces ge-0/2/0 encapsulation flexible-ethernet-services
```

```
set interfaces ge-0/2/0 unit 0 encapsulation vlan-bridge
set interfaces ge-0/2/0 unit 0 vlan-id 512
set interfaces ge-0/2/1 flexible-vlan-tagging
set interfaces ge-0/2/1 mtu 9192
set interfaces ge-0/2/1 encapsulation flexible-ethernet-services
set interfaces ge-0/2/1 unit 0 encapsulation vlan-bridge
set interfaces ge-0/2/1 unit 0 vlan-id 512
set bridge-domains bd1 vlan-id 10
set bridge-domains bd1 interface ge-0/2/1.0
set bridge-domains bd1 interface ge-0/2/0.0
```

Configuring VPLS Parameters on the MX Router (Provider Edge Router PE1)

```
set chassis fpc 0 pic 2 tunnel-services
set interfaces ge-0/2/1 flexible-vlan-tagging
set interfaces ge-0/2/1 mtu 9192
set interfaces ge-0/2/1 encapsulation vlan-vpls
set interfaces ge-0/2/1 unit 0 encapsulation vlan-vpls
set interfaces ge-0/2/1 unit 0 vlan-id 512
set interfaces ge-0/3/0 mtu 9192
set interfaces ge-0/3/0 unit 0 family inet address 192.0.2.1/24
set interfaces ge-0/3/0 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 198.51.100.1/32
set routing-options router-id 198.51.100.1
set routing-options autonomous-system 100
set protocols mpls interface ge-0/3/0.0
set protocols bgp group test type internal
set protocols bgp group test local-address 198.51.100.1
set protocols bgp group test family l2vpn signaling
set protocols bgp group test neighbor 198.51.100.2
set protocols ospf traffic-engineering
set protocols ospf reference-bandwidth 1g
set protocols ospf area 0.0.0.0 interface ge-0/3/0.0
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ldp interface ge-0/3/0.0 set protocols ldp interface lo0.0
set routing-instances vpls-pe1 instance-type vpls
set routing-instances vpls-pe1 interface ge-0/2/1.0
set routing-instances vpls-pe1 no-local-switching
set routing-instances vpls-pe1 route-distinguisher 198.51.100.1:101
set routing-instances vpls-pe1 vrf-target target:1:2
```

```
set routing-instances vpls-pe1 protocols vpls site-range 8
set routing-instances vpls-pe1 protocols vpls no-tunnel-services
set routing-instances vpls-pe1 protocols vpls site HUB site-identifier 1
set routing-instances vpls-pe1 protocols vpls vpls-id 1
set routing-instances vpls-pe1 protocols vpls neighbor 198.51.100.2
```

Configuring Benchmarking Test Parameters and VPLS Parameters on the MX104 Router (Provider Edge Router PE2)

```
set services rpm rfc2544-benchmarking tests test-name l2v-reflector source-mac-address
   00:00:5e:00:53:11
set services rpm rfc2544-benchmarking tests test-name l2v-reflector destination-mac-address
   00:00:5e:00:53:22
set services rpm rfc2544-benchmarking tests test-name l2v-reflector service-type elan
set services rpm rfc2544-benchmarking tests test-name l2v-reflector in-service
set services rpm rfc2544-benchmarking tests test-name l2v-reflector ip-swap
set services rpm rfc2544-benchmarking tests test-name l2v-reflector udp-tcp-port-swap
set services rpm rfc2544-benchmarking tests test-name l2v-reflector mode reflect
set services rpm rfc2544-benchmarking tests test-name l2v-reflector family vpls
set services rpm rfc2544-benchmarking tests test-name l2v-reflector reflect-etype 2048
set services rpm rfc2544-benchmarking tests test-name l2v-reflector direction egress
set services rpm rfc2544-benchmarking tests test-name l2v-reflector source-udp-port 200
set services rpm rfc2544-benchmarking tests test-name l2v-reflector destination-udp-port 200
set services rpm rfc2544-benchmarking tests test-name l2v-reflector test-interface ge-0/3/1.0
set interfaces ge-0/2/5 mtu 9192
set interfaces ge-0/2/5 unit 0 family inet address 203.0.113.1/24
set interfaces ge-0/2/5 unit 0 family mpls
set interfaces ge-0/3/1 flexible-vlan-tagging
set interfaces ge-0/3/1 mtu 9192
set interfaces ge-0/3/1 encapsulation vlan-vpls
set interfaces ge-0/3/1 unit 0 encapsulation vlan-vpls
set interfaces ge-0/3/1 unit 0 vlan-id 512
set interfaces ge-0/3/1 unit 0 family vpls filter input portmirror
set interfaces ge-0/3/1 unit 0 family vpls filter output portmirror
set interfaces ge-0/3/2 flexible-vlan-tagging
set interfaces ge-0/3/2 mtu 9192
set interfaces ge-0/3/2 encapsulation vlan-vpls
set interfaces ge-0/3/2 unit 0 encapsulation vlan-vpls
set interfaces ge-0/3/2 unit 0 vlan-id 512
set interfaces lo0 unit 0 family inet address 198.51.100.2/32
```

```
set forwarding-options port-mirroring input rate 1
set forwarding-options port-mirroring family vpls output interface ge-0/3/3.0
set forwarding-options port-mirroring family vpls output no-filter-check
set forwarding-options port-mirroring instance pm1 input rate 10000
set forwarding-options port-mirroring instance pm1 family vpls output interface ge-0/3/3.0
set routing-options router-id 198.51.100.2
set routing-options autonomous-system 100
set protocols mpls interface ge-0/2/5.0
set protocols bgp group test type internal
set protocols bgp group test local-address 198.51.100.2
set protocols bgp group test family l2vpn signaling
set protocols bgp group test neighbor 198.51.100.1
set protocols ospf traffic-engineering
set protocols ospf area 0.0.0.0 interface ge-0/2/5.0
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ldp interface ge-0/2/5.0
set protocols ldp interface lo0.0
set firewall family vpls filter portmirror term 1 then count pm1
set firewall family vpls filter portmirror term 1 then accept
set firewall family vpls filter portmirror term 1 then port-mirror
set routing-instances vpls-pe2 instance-type vpls
set routing-instances vpls-pe2 interface ge-0/3/1.0
set routing-instances vpls-pe2 interface ge-0/3/3.0
set routing-instances vpls-pe2 no-local-switching
set routing-instances vpls-pe2 route-distinguisher 198.51.100.2:102
set routing-instances vpls-pe2 vrf-target target:1:2
set routing-instances vpls-pe2 protocols vpls site-range 8
set routing-instances vpls-pe2 protocols vpls no-tunnel-services
set routing-instances vpls-pe2 protocols vpls site SPOKE site-identifier 2
set routing-instances vpls-pe2 protocols vpls vpls-id 1
set routing-instances vpls-pe2 protocols vpls neighbor 198.51.100.1
```

Configuring Throughput Benchmarking Test Parameters on the ACX Series Router (Initiator)

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the throughput test and reference the test profile in a unique test name. The test name defines the parameters for the throughput test to be performed on the ACX Series router.

To configure the throughput test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the first test profile—for example, tput—for the throughput test profile.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile tput
```

3. Configure the type of test to be performed as throughput, specify the packet size as 256 bytes, and define the theoretical maximum bandwidth for the test as 600000 Kbps. You can specify any value from 1 Kbps through 1,000,000 Kbps for the maximum bandwidth.

```
[edit services rpm rfc2544-benchmarking profiles test-profile tput]
user@host# set test-type throughput packet-size 256 bandwidth-kbps 600000
```

4. Enter the **up** command twice to go to the [**edit services rpm rfc2544-benchmarking**] level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile tput ]
user@host# up
user@host# up
```

5. Define a name for the throughput test—for example, tput-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name tput-test
```

6. Specify the name of the test profile, tput, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set test-profile tput
```

7. Configure the source and destination MAC addresses for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test to be E-LAN.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set ovlan-id 512 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress. Also, specify the source and destination UDP ports to be used in the UDP headers of the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set family bridge direction egress source-udp-port 200 destination-udp-port 400
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds, and specify the logical interface, ge-0/2/0.0, on which the RFC2544-benchmarking tests are run.

```
[edit services rpm rfc2544-benchmarking tests test-name tput-test]
user@host# set test-iterator-duration 250 test-interface ge-0/2/0.0
```

Configuring Back-to-Back Frames Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the back-to-back frames test and reference the test profile in a unique test name. The test name defines the parameters for the back-to-back frames test to be performed on the ACX Series router.

To configure the back-to-back frames test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the back-to-back test profile—for example, b2bt.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile b2bt
```

3. Configure the type of test to be performed as back-to-back frames, specify the packet size as 9104 bytes, and specify the theoretical maximum bandwidth for the test as 600000 Kbps. You can specify any value from 1 Kbps through 1,000,000 Kbps as the maximum bandwidth.

```
[edit services rpm rfc2544-benchmarking profiles test-profile b2bt]
user@host# set test-type back-toback-frames packet-size 9104 bandwidth-kbps 600000
```

4. Enter the **up** command twice to go to the [**edit services rpm rfc2544-benchmarking**] level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile b2bt ]
user@host# up
user@host# up
```

5. Define a name for the back-to-back frames test—for example, b2bt-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name b2bt-test
```

6. Specify the name of the test profile, b2bt, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set test-profile b2bt
```

7. Configure the source and destination MAC addresses for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test as E-LAN.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set ovlan-id 512 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set family bridge direction egress
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds. Also, specify the logical interface, ge-0/2/0.0, on which the RFC2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name b2bt-test]
user@host# set test-iterator-duration 10 test-interface ge-0/2/0.0
```

Configuring Latency Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the latency test and reference the test-profile in a unique test-name. The test-name defines the parameters for the latency test to be performed on the initiator (ACX Series router).

To configure the latency test parameters on the initiator:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the latency test profile—for example, lty.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile lty
```

3. Configure the type of test to be performed as latency, specify the packet size of the test packet as 1024, and specify the maximum bandwidth for the test in Kbps, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# set test-profile lty test-type latency packet-size 1024 bandwidth-kbps 600000
```

4. Enter the **up** command twice to go to the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile lty]
user@host# up
user@host# up
```

5. Define a name for the latency test—for example, lty-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name lty-test
```

6. Specify the name of the test profile, lty, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set test-profile lty
```

7. Configure the source and destination MAC addresses for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID for the test frames and specify the service type under test.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set ovlan-id 512 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress. Also, specify the source and destination UDP port to be used in the UDP headers of the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set family bridge direction egress source-udp-port 200 destination-udp-port 400
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds. Also, specify the logical interface, ge-0/2/0.0, on which the RFC2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name lty-test]
user@host# set test-iterator-duration 10 test-interface ge-0/2/0.0
```

Configuring Frame Loss Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following configuration requires you to configure a test profile for the frame loss test and reference the test-profile in a unique test-name. The test-name defines the parameters for the frame loss test to be performed on the ACX Series router.

To configure the frame loss test parameters on the ACX Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the frame loss test profile—for example, frloss.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile frloss
```

3. Configure the type of test performed as frame loss, specify the packet size of the test packet, and define the maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# set test-profile frloss test-type frame-loss packet-size 1600 bandwidth-kbps 600000
```

4. Enter the **up** command to go to the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles ]
user@host# up
```

5. Define a name for the frame loss test—for example, frloss-test. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name frloss-test
```

6. Specify the name of the test profile, frloss, to be associated with the test name.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set test-profile frloss
```

7. Configure the source and destination MAC address for the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

8. Configure the outer VLAN ID, priority, and the canonical format indicator (cfi) value for the test frames. Together, the four added bytes, priority (3 bits) and canonical format indicator (1 bit) form the VLAN tag. Also, specify the service type under test.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set ovlan-id 512 ovlan-priority 7 ovlan-cfi 1 service-type elan
```

9. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set mode initiate-and-terminate
```

10. Configure the family type, **bridge**, for the benchmarking test and specify the direction, egress. Also, specify the source and destination UDP port to be used in the UDP headers of the test packet.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set family bridge direction egress source-udp-port 200 destination-udp-port 400
```

11. Specify the duration of each iteration in seconds, with a value from 10 seconds to 1,728,000 seconds. Also, specify the logical interface, ge-0/2/1.0, on which the RFC2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name frloss-test]
user@host# set test-iterator-duration 30 test-interface ge-0/2/0.0
```

12. Enter the **exit** command to go to the [edit] hierarchy level.

```
[edit services rpm rfc2544-benchmarking tests test-name test4 ]
user@host# exit
```

Configuring Other Benchmarking Test Parameters on the ACX Series Router

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see Using the CLI Editor in Configuration Mode in the CLI User Guide.

To configure the interface and bridge domain on the ACX Series router:

1. Configure the Layer 2 NNI interface on which the tests must be run from the [**edit**] hierarchy level.

```
[edit]
user@host# edit interfaces ge-0/2/1
```

2. Configure flexible VLAN tagging for the transmission of untagged frames or 802.1Q single-tagged and dual-tagged frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-0/2/1]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation flexible-ethernet-services
```

3. Configure a logical unit for the interface, specify the encapsulation, and configure the VLAN ID on the logical interfaces.

```
[edit interfaces ge-0/2/1]
user@host# set unit 0 encapsulation vlan-bridge vlan-id 512
```

4. Configure the Layer 2 UNI interface.

```
[edit]
```

```
user@host# edit interfaces ge-0/2/0
```

5. Configure flexible VLAN tagging for transmission of non-tagged frames or 802.1Q single-tag and dual-tag frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-0/2/0]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation flexible-ethernet-services
```

6. Configure a logical unit for the interface and specify the encapsulation and configure the VLAN ID on the logical interfaces.

```
[edit interfaces ge-0/2/0]
user@host# set unit 0 encapsulation vlan-bridge vlan-id 512
```

7. Configure the bridge domain, bd1, and specify the VLAN ID associated with the bridge domain and the associated interfaces from the [**edit**] hierarchy level.

```
[edit ]
user@host# set bridge-domains bd1 vlan-id 10 interface ge-0/2/1.0
user@host# set bridge-domains bd1 vlan-id 10 intreface ge-0/2/0.0
```

Configuring the VPLS Parameters on the MX Series Router (PE1)

Step-by-Step Procedure

The following configuration requires you to enable a simple VPLS topology between the PE1 and PE2 routers. In this example, PE1 is a MX Series router. On the PE1 router, configure the tunnel services interface and prepare the router for VPLS by configuring the BGP, MPLS, OSPF protocols.

To configure the VPLS parameters on the MX Series router:

1. Configure tunnel services.

```
[edit ]
user@host# set chassis fpc 0 pic 2 tunnel-services
```

2. Configure the VPLS VLAN encapsulation on the router.

```
[edit interfaces]
```

```
user@host# set interfaces ge-0/2/1 flexible-vlan-tagging
user@host# set interfaces ge-0/2/1 mtu 9192
user@host# set interfaces ge-0/2/1 encapsulation vlan-vpls
user@host# set interfaces ge-0/2/1 unit 0 encapsulation vlan-vpls
user@host# set interfaces ge-0/2/1 unit 0 vlan-id 512
```

3. Configure the routing interface and the loopback interface on the router.

```
[edit interfaces]
user@host# set interfaces ge-0/3/0 mtu 9192
user@host# set interfaces ge-0/3/0 unit 0 family inet address 192.0.2.1/24
user@host# set interfaces ge-0/3/0 unit 0 family mpls
user@host# set interfaces lo0 unit 0 family inet address 198.51.100.1/32
```

4. Configure the routing options on the router.

```
[edit routing-options]
user@host# set routing-options router-id 198.51.100.1
user@host# set routing-options autonomous-system 100
```

5. Configure MPLS on the router to advertise the Layer 2 VPN interface that communicates with the PE2 router.

```
[edit protocols]
user@host# set protocols mpls interface ge-0/3/0.0
```

6. Configure BGP as the signaling protocol on the router to enable carrying of Layer 2 VPLS messages.

```
[edit protocols]
user@host# set protocols bgp group test type internal
user@host# set protocols bgp group test local-address 198.51.100.1
user@host# set protocols bgp group test family l2vpn signaling
user@host# set protocols bgp group test neighbor 198.51.100.2
```

7. Configure OSPF on the router to enable exchange of routing information.

```
[edit protocols]
user@host# set protocols ospf traffic-engineering
user@host# set protocols ospf reference-bandwidth 1g
```

```
user@host# set protocols ospf area 0.0.0.0 interface ge-0/3/0.0
user@host# set protocols ospf area 0.0.0.0 interface lo0.0
```

8. Configure LDP on the router to enable LDP for all connections

```
[edit protocols]
user@host# set protocols ldp interface ge-0/3/0.0
user@host# set protocols ldp interface lo0.0
```

9. Create and configure the VPLS routing interface.

```
[edit routing instances vpls-instance]
user@host# set routing-instances vpls-pe1 instance-type vpls
user@host# set routing-instances vpls-pe1 interface ge-0/2/1.0
user@host# set routing-instances vpls-pe1 no-local-switching
user@host# set routing-instances vpls-pe1 route-distinguisher 198.51.100.1:101
user@host# set routing-instances vpls-pe1 vrf-target target:1:2
user@host# set routing-instances vpls-pe1 protocols vpls site-range 8
user@host# set routing-instances vpls-pe1 protocols vpls no-tunnel-services
user@host# set routing-instances vpls-pe1 protocols vpls site HUB site-identifier 1
user@host# set routing-instances vpls-pe1 protocols vpls vpls-id 1
user@host# set routing-instances vpls-pe1 protocols vpls neighbor 198.51.100.2
```

Configuring Benchmarking Test Parameters on the MX104 Router (Reflector)

Step-by-Step Procedure

The following configuration requires you to configure a unique test-name for the benchmarking test on the MX104 Series router. The test-name defines the parameters for the benchmarking test to be performed. Because the test interface and test MAC addresses are the same, you can create a single test configuration at the reflector (MX104).

To configure the benchmarking test parameters on the MX104 Series router:

1. In configuration mode, at the [**edit**] hierarchy level, configure a real-time performance monitoring service (RPM) instance and an RFC2544-based benchmarking test for the RPM instance.

```
[edit]
user@host# edit services rpm rfc2544-benchmarking
```

2. Define a name for the test—for example, l2v-reflector. The test name can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking ]
user@host# edit tests test-name l2v-reflector
```

3. Specify the source and destination MAC addresses of the test packet.

```
[edit services rpm rfc2544-benchmarking test-name l2v-reflector]
user@host# set source-mac-address 00:00:5e:00:53:11 destination-mac-address 00:00:5e:00:53:22
```

4. Specify the service type under test and the mode in which the test is executed, which is in-service, at the reflector. Also, specify if the IP address, TCP and UDP port must be swapped.

```
[edit services rpm rfc2544-benchmarking test-name l2v-reflector]
user@host# set service-type elan in-service ip-swap udp-tcp-port-swap
```

5. Specify the mode which is reflect at the reflector.

```
[edit services rpm rfc2544-benchmarking test-name l2v-reflector]
user@host# set mode reflect
```

6. Configure the family type, **vpls**, specify the direction, egress, and specify the protocol being transported in the Ethernet frame, for the benchmarking test. Also, specify the source and destination UDP ports and specify the logical interface, ge-0/3/1.0, on which the RFC2544-based benchmarking test is being run.

```
[edit services rpm rfc2544-benchmarking tests test-name l2v-reflector]
user@host# set family vpls direction egress source-udp-port 200 destination-udp-port 200 test-interface
   ge-0/3/1.0
```

Configuring Other Benchmarking Test Parameters on the MX104 Router (Reflector)

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see Using the CLI Editor in Configuration Mode in the CLI User Guide.

To configure the interface and bridge domain on the MX104 Series router:

1. Configure the Layer 2 NNI interface on which the tests must be run.

```
[edit]
```

```
user@host# edit interfaces ge-0/3/1.0
```

2. Configure flexible VLAN tagging for transmission of untagged frames or 802.1Q single-tagged and dual-tagged frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-0/3/1.0]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation vlan-vpls
```

3. Configure a logical unit for the interface, specify the encapsulation, and configure the VLAN ID on the logical interface.

```
[edit interfaces ge-0/3/1.0]
user@host# set unit 0 encapsulation vlan-vpls vlan-id 512
```

4. Configure the Layer 2 UNI interface.

```
[edit]
user@host# edit interfaces ge-0/3/2.0
```

5. Configure flexible VLAN tagging for transmission of untagged frames or 802.1Q single-tagged and dual-tagged frames on the logical interface. You can also specify the maximum transmission unit (MTU) size for the interface and the encapsulation.

```
[edit interfaces ge-0/3/2.0]
user@host# set flexible-vlan-tagging mtu 9192 encapsulation vlan-vpls
```

6. Configure a logical unit for the interface, specify the encapsulation, and configure the VLAN ID on the logical interfaces.

```
[edit interfaces ge-0/3/2.0]
user@host# set unit 0 encapsulation vlan-vpls vlan-id 512
```

7. //**Configure the bridge domain, bd1, and specify the VLAN ID associated with the bridge domain, and the associated interfaces from the [**edit**] hierarchy level.

```
[edit ]
```

```
user@host# set bridge-domains bd1 vlan-id 500 interface ge-1/1/6.0
user@host# set bridge-domains bd1 vlan-id 500 intreface ge-1/1/5.0 **//
```

8. Start the benchmarking test on the reflector.

```
user@host> test services rpm rfc2544-benchamrking test l2v-reflector start
```

After the test is successfully completed at the initiator, you can stop the test at the reflector by entering the **test services rpm rfc2544-benchmarking test l2v-reflector stop** command.

Configuring VPLS Parameters on the MX104 Router (Reflector)

Step-by-Step Procedure

The following configuration requires you to enable a simple VPLS topology between the PE1 and PE2 routers. In this example, PE2 is a MX104 Series router. On the PE2 router, configure the tunnel services interface and prepare the router for VPLS by configuring the BGP, MPLS, OSPF protocols to complement the configuration on PE1.

1. Configure tunnel services.

```
[edit ]
user@host# set chassis fpc 0 pic 2 tunnel-services
```

2. Configure the VPLS VLAN encapsulation on the router.

```
[edit interfaces]
user@host# set interfaces ge-0/2/5 flexible-vlan-tagging
user@host# set interfaces ge-0/2/5 mtu 9192
user@host# set interfaces ge-0/2/5 encapsulation vlan-vpls
user@host# set interfaces ge-0/2/5 unit 0 encapsulation vlan-vpls
user@host# set interfaces ge-0/2/5 unit 0 vlan-id 512
```

3. Configure the routing interface and the loopback interface on the router.

```
[edit interfaces]
user@host# set interfaces ge-0/3/0 mtu 9192
user@host# set interfaces ge-0/3/0 unit 0 family inet address 192.0.2.1/24
user@host# set interfaces ge-0/3/0 unit 0 family mpls
user@host# set interfaces lo0 unit 0 family inet address 198.51.100.1/32
```

4. Configure the routing options on the router.

```
[edit routing-options]
user@host# set routing-options router-id 198.51.100.1
user@host# set routing-options autonomous-system 100
```

5. Configure MPLS on the router to advertise the Layer 2 VPN interface that communicates with the PE1 router.

```
[edit protocols]
user@host# set protocols mpls interface ge-0/2/5.0
```

6. Configure BGP as the signaling protocol on the router to enable carrying of Layer 2 VPLS messages.

```
[edit protocols]
user@host# set protocols bgp group test type internal
user@host# set protocols bgp group test local-address 198.51.100.1
user@host# set protocols bgp group test family l2vpn signaling
user@host# set protocols bgp group test neighbor 198.51.100.2
```

7. Configure OSPF on the router to enable exchange of routing information.

```
[edit protocols]
user@host# set protocols ospf traffic-engineering
user@host# set protocols ospf reference-bandwidth 1g
user@host# set protocols ospf area 0.0.0.0 interface ge-0/2/5.0
user@host# set protocols ospf area 0.0.0.0 interface lo0.0
```

8. Configure LDP on the router to enable LDP for all interfaces.

```
[edit protocols]
user@host# set protocols ldp interface ge-0/2/5.0
user@host# set protocols ldp interface lo0.0
```

9. Create and configure the VPLS routing interface.

```
[edit routing instances vpls-instance]
user@host# set routing-instances vpls-pe2 instance-type vpls
user@host# set routing-instance vpls-pe2 interface ge-0/3/1.0
```

```
user@host# set routing-instances vpls-pe2 no-local-switching
user@host# set routing-instances vpls-pe2 route-distinguisher 198.51.100.1:101
user@host# set routing-instances vpls-pe2 vrf-target target:1:2
user@host# set routing-instances vpls-pe2 protocols vpls site-range 8
user@host# set routing-instances vpls-pe2 protocols vpls no-tunnel-services
user@host# set routing-instances vpls-pe2 protocols vpls site SPOKE site-identifier 1
user@host# set routing-instances vpls-pe2 protocols vpls vpls-id 1
user@host# set routing-instances vpls-pe2 protocols vpls neighbor 198.51.100.2
```

Results

In configuration mode, confirm your configuration on the ACX Series router, the MX Series router, and the MX104 Series router by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Benchmarking Test Parameters on the ACX Series router:

```
[edit interfaces]
  ge-0/2/0 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation flexible-ethernet-services;
      unit 0 {
encapsulation vlan-bridge;
          vlan-id 512;
        }
  }
ge-0/2/1 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation flexible-ethernet-services;
      unit 0 {
encapsulation vlan-bridge;
          vlan-id 512;
             }
  }

[edit bridge-domains]
 bd1 {
    vlan-id 600;
    interface ge-0/2/1.0;
    interface ge-0/2/0.0;
}
```

```
  [edit services rpm]
  rfc2544-benchmarking {
      profiles {
          test-profile tput {
              test-type throughput
              packet-size 256;
              bandwidth-kbps 600000;
          }
test-profile b2bt {
              test-type back-back-frames
              packet-size 9104;
             bandwidth-kbps 600000;
          }
test-profile lty {
              test-type latency
              packet-size 1024;
              bandwidth-kbps 600000;
          }
test-profile frloss {
              test-type frameloss
              packet-size 1600;
              bandwidth-kbps 6000000;
          }
              }

      tests {
          test-name tput-test {
              interface ge-0/2/0.0;
              test-profile tput;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 512;
              service-type elan;
              family bridge;
  direction egress;
  source-udp-port 200;
  destination-udp-port 400;
  test-iterator-duration 250;
          }
test-name b2b-test {
              interface ge-0/2/0.0;
              test-profile b2bt;
```

```
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 512;
              service-type elan;
              family bridge;
  direction egress;
  destination-udp-port 400;
  test-iterator-duration 10;
          }
test-name lty-test {
              interface ge-0/2/0.0;
              test-profile lty;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 512;
              service-type elan;
              family bridge;
  direction egress;
  source-udp-port 200;
  destination-udp-port 400;
  test-iterator-duration 10;
          }
test-name frloss-test {
              interface ge-0/2/0.0;
              test-profile frloss;
              mode initiate,terminate;
              source-mac-address 00:00:5e:00:53:11;
             destination-mac-address 00:00:5e:00:53:22;
  ovlan-id 512;
              service-type elan;
              family bridge;
  direction egress;
  source-udp-port 200;
  destination-udp-port 400;
  test-iterator-duration 30;
          }
      }
  }
```

VPLS Parameters on the MX Series router:

```
[edit routing-instances]
```

```
vpls-instance {
 instance-type vpls;
interface ge-0/2/1.0;
route-distinguisher 198.51.100.1:101;
vrf-target target:1:2;
}
protocols {
vpls {
vpls-id 1;
neighbor 198.51.100.2;
site-range 8;
no-tunnel-services;
site HUB {
site-identifier 1;
}
```

Benchmarking Test Parameters and VPLS Parameters on the MX104 Series router:

```
[edit interfaces]
  ge-0/3/1 {
      flexible-vlan-tagging;
 mtu 9192;
encapsulation vlan-vpls;
      unit 0 {
encapsulation vlan-vpls;
          vlan-id 512;
        }
  }
ge-0/2/5 {
      flexible-vlan-tagging;
 mtu 9192;
 unit 0 {
family inet address 203.0.113.1/24;
          family mpls;
        }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
      tests {
          test-name l2v-reflector {
              interface ge-0/3/1.0;
              source-mac-address 00:00:5e:00:53:11;
```

```
            destination-mac-address 00:00:5e:00:53:22;
  mode reflect;
              service-type elan;
   in-service;
   ip-swap;
   udp-tcp-port swap;
               family vpls;
   reflect-etype 2048;
   direction egress;
   source-udp-port 200;
   destination-udp-port 200;
            }
       }
  }

[edit routing-instances]
vpls-instance {
 instance-type vpls;
interface ge-0/3/1;
route-distinguisher 198.51.100.2:102;
vrf-target target:1:2;
}
protocols {
vpls {
vpls-id 1;
neighbor 198.51.100.1;
site-range 8;
no-tunnel-services;
site SPOKE {
site-identifier 2;
}
```

After you have configured the device, enter the **commit** command, in configuration mode.

Verifying the Results of the Benchmarking Test for Layer 2 ELAN Services Using VPLS

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between the ACX Series router and the MX104 Series router.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between the ACX Series router and the MX104 Series router.

Action

In operational mode, enter the **show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Example: Configuring RFC2544-Based Benchmarking Tests on an MX104 Router for Layer 2 E-LAN Services in Bridge Domains | **701**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- Supported RFC2544-Based Benchmarking Statements on MX Series Routers | **657**

CHAPTER 17

Configuring RFC 2544-Based Benchmarking Tests on ACX Series

IN THIS CHAPTER

RFC 2544-Based Benchmarking Tests Overview

RFC 2544 defines a series of tests that can be used to describe the performance characteristics of network interconnecting devices. RFC2544-based benchmarking tests methodology can be applied to a single device under test (DUT), or a network service (set of devices working together to provide end-to-end service). When applied to a service, the RFC2544 test results can characterize the Service-Level-Agreement (SLA) parameters.

RFC 2544 tests are performed by transmitting test packets from a device that functions as the generator or the initiator. These packets are sent to a device that functions as the reflector, which receives and returns the packets back to the initiator.

ACX Series routers support RFC 2544 tests to measure the following:

- Throughput
- Latency

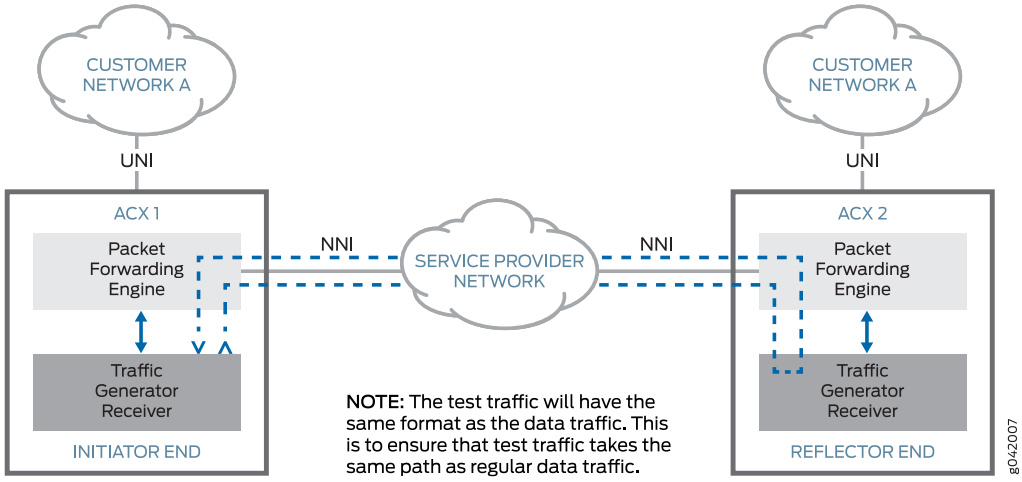
- Frame loss rate
- Back-to-back frames

With embedded RFC 2544, an ACX Series router can be configured as an initiator and another ACX Series router as a reflector.

> **NOTE:** ACX5048 and ACX5096 routers can be configured only as a reflector.

Figure 56 on page 764 shows the components, role of intiator and reflector, and the flow of test packets in an RFC 2544-based benchmarking test.

Figure 56: RFC 2544-Based Benchmarking Test Methodology



To run RFC 2544-based tests, you need a router to generate service test traffic and a router to reflect the service test traffic back. You need to:

1. Identify two service endpoints between which the RFC2544-based test needs to be run.
2. Configure the reflector end and start reflection.
3. Configure the initiator end and initiate the test.
4. Review the results after the test is complete. Test results are reported in a specific format.

On ACX Series routers, you can run the following RFC 2544-based performance measurement tests:

- Throughput test:
  - Sends a specific number of frames at a specified rate from the initiator through the network service or a DUT. The test starts with a user-configured theoretical maximum rate.
  - Counts the number of transmitted frames and the number of received frames.

  - If the number of frames received is less than those transmitted, the test is repeated with a 50 percent reduced frame rate.
  - Throughput is the maximum rate at which the count of test frames received is equal to the number of test frames transmitted through the network service.

  You can repeat throughput tests for different frame sizes.

- Latency test:

  > **NOTE:** To run latency test, you need to determine the throughput for DUT or a network service at each of the specified frame sizes.

  - Starts with a stream of frames at a particular frame size through the DUT at the determined throughput rate.
  - Sends an identifying tag in one frame after 60 seconds and calculate the latency when the frame with the same tag is received by the initiator.
  - Is repeated for at least 20 times with the reported latency value being the average of the recorded values.

  You can repeat latency tests for different frame sizes.

- Frame loss rate test:
  - Involves sending a specific number of frames at a specified rate through the DUT or a network service to be tested and counting the frames that are transmitted.
  - Calculates frame loss rate at each point using the equation:
    ((input_count - output_count) x 100) / input_count.
  - Runs a trial for the frame rate that corresponds to 100 percent of the configured maximum theoretical rate.
  - Is repeated for the frame rate that corresponds to 90 percent of the maximum rate used and then for 80 percent of the maximum rate until a certain trial result shows no lost frames.

  You repeat the frame loss rate tests for different frame sizes.

- Back-to-back frames test:
  - Involves sending a burst of frames with minimum interframe gaps through the DUT or a network service and counting the number of frames forwarded.
  - Is rerun with an increased length of burst frames if the count of transmitted frames is equal to the number of frames forwarded.
  - Is rerun with a reduced length of burst frames if the count of forwarded frames is less than the number of frames transmitted.

The back-to-back value is the number of frames in the longest burst that the DUT or a network service can handle without the loss of any frames.

You can repeat back-to-back frame tests for different frame sizes.

> **NOTE:** In ACX Series routers, RFC 2544 tests are supported only for E-LINE, ELAN, and EVPL services.
>
> ACX5048 and ACX5096 routers supports only E-LINE services. ACX5048 and ACX5096 routers do not support family inet based reflection. Family bridge and family CCC are supported.

ACX5448 router supports:

- RFC2544 egress Layer 2 reflection functionality for family bridge.
- Multiple RFC2544 reflection sessions.
- Reflection on 1G/10G/40G/Ch10G/Ch25G/100G ports.
- Ethernet Layer 2 frames to carry IP/UDP packets for RFC2544 reflection.

ACX5448 router do not support the following RFC2544 features:

- Any interface in the bridge domain matching the bridge VLAN identifier is not supported.
- Multiple simultaneous sessions with multiple VLAN bridges are not supported.
- Multiple test sessions cannot exceed 100G bandwidth.
- IPv6 reflection.
- IPV6 filter support to identify the loopback stream.
- RFC 2544 reflection functionality for family **ccc** (PWE reflection) and family **inet** (Layer 3 IPV4 reflection).
- Reflection without MAC swap and MAC overwrite is not supported.
- Reflection on E-LINE and E-LAN services.

RELATED DOCUMENTATION

Layer 2 and Layer 3 RFC 2544-Based Benchmarking Test Overview | **767**

Configuring RFC 2544-Based Benchmarking Tests | **770**

show services rpm rfc2544-benchmarking | **1371**

show services rpm rfc2544-benchmarking test-id | **1377**

Layer 2 and Layer 3 RFC 2544-Based Benchmarking Test Overview

In ACX Series routers, RFC 2544-based benchmark tests can be run to measure the performance characteristic of the E-LINE, E-LAN, and EVPL services.

> **NOTE:** ACX5048 and ACX5096 routers supports only E-LINE services. ACX5048 and ACX5096 routers do not support family inet based reflection. Family bridge and family CCC are supported.

- You can configure the test on the following underlying services:
  - Between two IPv4 endpoints—In this mode, the generator sends test packets to user-configured IP destination or UDP port (which is of the reflector).
  - Between two user-to-network interfaces (UNIs) of Ethernet Virtual Connection (EVC), Ethernet Private Line (EPL, also called E-LINE), Ethernet Virtual Private Line (EVPL), EVC (EPL, EVPL)—One end is configured as the generator or initiator and the other end acts as the reflector. The generator receives the test packets that are returned from the reflector and computes the test results.

    > **NOTE:** Benchmarking tests are not supported for IPv6-based services.

- You cannot perform multiple simultaneous RFC 2544-based benchmarking tests on the same pseudowire.
- Interoperation of the RFC 2544 benchmarking tests with other third-party customer premises equipment (CPE) that provides embedded or dedicated benchmarking test capability is not supported.
- Fragmented test-frames and one-way measurements of frames are not supported. You must configure one end or the source device to initiate and terminate test frames and the other end or the destination device to reflect the received frames back to the initiator.
- RFC 2544 generator and reflector are supported with testing bandwidth up to 1 Gbps. ACX5048 and ACX5096 routers supports test bandwidth of up to 40 Gbps.
- The test session is supported in out-of-service mode for the underlying service. You must not transmit any traffic to the UNI port, configured as a generator or a reflector, that is being tested during the duration of the test. However, other services that are not configured for the testing session are not impacted.
- Devices embedded with benchmarking test capabilities (generators and reflectors) interoperate with other Juniper Networks devices that support the RFC 2544-based generator or reflector functionality.
- RFC 2544 generator traffic undergoes the same traffic classifier and policer or shaper processing as the ingress customer traffic from the UNI port.

- RFC 2544 generator produces a report with clear details of pass or fail for each critical testing metric, based on the configured thresholds.
- The testing packets can be configured and the format of the packet depends on the underlying service on which the test is configured. For IP-based service, the IP or port values can be configured. For Ethernet-based service, unicast untagged or VLAN ID-tagged dot1p formats (IEEE 802.1p or packet classification Layer 2 headers) are supported. The Ethernet destination address and source address that you configured are used.
- You can run RFC 2544 benchmarking **inet** tests on Layer 3 VPN or virtual router.
- For an **inet** service, each test session needs to use a unique UDP port. On the initiator device, the source UDP port that you specify by using the **source-udp-port** statement must be unique and not used by other UDP services that terminate at the initiator. On the reflector device, the UDP port of the destination to be used in the UDP header for the generated frames by using the **destination-udp-port** statement must be unique and not used by other UDP services that terminate at the reflector.
- You must start the test on the router that operates as the reflector before you start the test on router that functions as the initiator.
- You must configure the size of the test packet based on the configured MTU of the packets.
- For computation of the test results for a user-to-network interface (UNI) or ingress direction of an Ethernet pseudowire service, the customer edge (CE) device that is configured as a reflector for **inet** must have the reflected destination address resolved using ARP or a statically configured route must be present on the CE device to connect to the initiator.
- For benchmarking tests on the UNI direction of an Ethernet pseudowire service, if reflection mode is configured, you must configure a static ARP entry. Otherwise, the tests fail when test frames on the UNI interface are reflected. ARP resolution does not enable a successful reflection of test frames for UNI interfaces.
- For a CCC family and with the test performed in the egress or network-to-network interface (NNI) direction, the tests stop on the initiator and reflector when the pseudowire goes down.
- For an RFC 2544 test that is run in the egress or network-to-network interface (NNI) direction of an Ethernet service for a CCC family, the ingress features are not applied.
- In ACX5048 and ACX5096 routers, for a CCC family, the pseudowire has to be opened prior to the start of the RFC 2544 test and during the course of the test.
- The configured packet size denotes the untagged packet size. Any additional VLAN in the payload causes the packet length to be increased correspondingly.
- For an **inet** service, if you configure an interface on an initiator for the RFC 2544-based benchmarking test to be run without specifying the source IPv4 address for the test frames, the primary IP address of the interface is used for the test frames. If the primary IP address is not configured, the first IPv4 address of the interface is used. Similarly, for an unnumbered interface on an initiator on which the RFC 2544 test is run, the primary or the first IP address of the donor loopback interface is retrieved and used in

the test frames. You must explicitly configure the source IPv4 address for the test frames by using the **source-ipv4-address** statement if you want a particular address to be used.

- RFC 2544 test generates packets for performance benchmarking testing. The packets can be destined for known or unknown unicast MAC addresses, and they can be either tagged or untagged frames. UDP/IP packet is used as the frame payload. Refer to "Configuring RFC 2544-Based Benchmarking Tests" on page 770 for the frame fields that can be configured.
- Supported outer TPIDs for tagged frames are 0x8100, 0x88a8, 0x9100, and 0x9200.
- RFC 2544 benchmark tests can be run in **out-of-service** and in **in-service** modes.

  NOTE: In **out-of-service** mode, while the test is running, all the data traffic sent to and from the UNI port under test on the service is interrupted. Control protocol packets are not interrupted.

  In **in-service** mode, while the test is running, only the data traffic corresponding to the test session is interrupted, rest of the data traffic flow sent to and from the UNI port under test on the service are not affected. Control protocol packets are not interrupted.

- The source MAC address, destination MAC address, and the UNI port under test configured uniquely identifies the RFC 2544 benchmark test session (or test stream).
- You can run only one test at a time. Multiple simultaneous tests cannot be run at a time.
- The maximum theoretical test bandwidth supported by ACX Series routers for RFC 2544 test initiator or reflector is 1 Gpbs. On AC5048 and ACX5096 routers, the maximum theoretical test bandwidth supported for RFC 2544 reflector is 40 Gbps.
- RFC 2544 tests can be run with different frame sizes. In ACX Series routers, the supported frame sizes are 64, 68, 72, 128, 256, 512, 768, 1024, 1280, 1518, 1522, 1600, 1728, 2496, 3584, 4016, 9104, and 9136 bytes.
- The minimum mandated duration for RFC 2544 benchmark tests to run is 10 seconds.
- The test uses round-trip traffic for performance measurement.
- A history of the test results is stored in memory.
- The test results can be copied to the local file system or a remote file system, optionally.

  NOTE: RFC 2544 test is not supported to compute the performance attributes of multicast or broadcast traffic streams.

RELATED DOCUMENTATION

- RFC 2544-Based Benchmarking Tests Overview | 763
- Configuring RFC 2544-Based Benchmarking Tests | 770
- show services rpm rfc2544-benchmarking | 1371
- show services rpm rfc2544-benchmarking test-id | 1377

Configuring RFC 2544-Based Benchmarking Tests

IN THIS SECTION

To configure a RFC 2544 benchmark test on an initiator, you must first configure a **test-profile** and reference the **test-profile** in a unique **test-name**. The **test-name** defines the parameters for the tests to be performed.

To configure a **test-profile**, include the **test-profile *profile-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level. Test profile is applicable only for initiator.

To configure a **test-name**, include the **test-name *test-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level.

To configure Ethernet loopback as the test mode on a logical interface, include the **Ethernet-loopback** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level.

NOTE: The **test-profile** is not required while configuring the reflector for RFC 2544 test.

Table 106 on page 770 lists the parameters for configuring test-profile at initiator.

Table 106: Parameters for test-profile Configuration

Parameters	Description
test-type	RFC 2544 test type (**throughput | latency | frame-loss | back-back-frames**).

Table 106: Parameters for test-profile Configuration *(continued)*

Parameters	Description
packet-size	Size of the test packet. The valid packet sizes are 64, 68, 72, 128, 256, 512, 768, 1024, 1280, 1518, 1522, 1600, 1728, 2496, 3584, 4016, 9104, and 9136 bytes.
bandwidth-kbps	Define the maximum bandwidth limit, in kilobits per second (kbps). Range: 1,000 kpbs through 1,000,000 kbps.
step-percent	Specify the step percentage for frame-loss tests. Default: 10 percent Range: 1 through 100 percent

Table 107 on page 771 lists the parameters for configuring a test-name at initiator and reflector.

Table 107: Parameter for test-name configuration

Parameters	Description
check-test-interface-mtu	When the **check-test-interface-mtu** parameter is configured, the parameter validates the MTU size of the test packets with the MTU size configured on the interface and the following would be the behavior for initiator and reflector modes: • On the initiator, if the MTU size of the test packet is larger than the MTU size configured on the interface, then the RFC2544-based benchmarking test fails to start. • On the reflector, if the test packets coming to the reflector does not confirm to the MTU size configured on the interface, then these test packets do not get reflected and are dropped.
destination-ipv4-address	Specify the destination IPv4 address. This parameter is mandatory when family **inet** is specified and optional when family **ccc** is specified. If a value is not specified, then by default 192.168.1.20 is used. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
destination-mac-address	Specify the destination MAC address. For example, 0011.2233.4455. This parameter cannot be used when family **inet** is specified. This parameter is optional when family **ccc** is specified. If not specified, then the default value of 0x00:0x11:0xAE:0x92:0x2F:0x28 is used.

Table 107: Parameter for test-name configuration *(continued)*

Parameters	Description
destination-udp-port	Specify the destination UDP port number for the test frames. Default: 4041. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
direction	Specify the test direction (**egress | ingress**). This parameter is valid only when family **ccc** and **bridge**. This parameter is mandatory for mode **ethernet-loopback**
disable-signature-check	Disable signature verification on the received test frames.
dscp-code-points	Specify the value of the Differentiated Services (DiffServ) field. For example, 001111. If a value is not specified, then '0' is used in IP header. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
family	Configure the test family (**bridge | ccc | inet**). This parameter is mandatory for mode **ethernet-loopback**
forwarding-class	Specify the forwarding class to be used for test frames.
halt-on-prefix-down	If specified, a prefix that moves to the down state causes the corresponding tests to be stopped. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
ignore-test-interface-state	When the **ignore-test-interface-state** parameter is configured for RFC2544 benchmarking tests, the test continues to run even if there are any occurrences of interface up or down events. This is applicable to both initiator and reflector test modes.
in-service	If specified, only the data traffic corresponding to the test session is interrupted, rest of the data traffic flow sent to and from the UNI port under test on the service are not affected. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
ivlan-cfi	CFI bit used in the inner VLAN tag. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.

Table 107: Parameter for test-name configuration *(continued)*

Parameters	Description
ivlan-id	Configure inner VLAN ID for the test frames. This parameter is valid only for family **ccc** mode. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
ivlan-priority	Configure the priority value for the IEEE 802.1p bit in the inner VLAN tag. Range: 0 through 7. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
mode	Specify the test mode (**ethernet-loopback**, **initiate-and-terminate**, or **reflect**). • **ethernet-loopback**—Test frames are loopbacked to the measuring device after the source MAC address and the destination MAC addresses are swapped. • **initiate-and-terminate**—Test frames are initiated and terminated at the same end. If you specify this mode, then a reflector should be configured on the peer end to bring back the test frames. • **reflect**—Test frames are reflected on the chosen service.
outer-tag-protocol-id	TPID to be used in the outer VLAN tag. Supported values are 0x8100, 0x88a8, 0x9100, 0x9200. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
ovlan-cfi	CFI bit used in the outer VLAN tag. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
ovlan-id	Configure the outer VLAN ID for the test frames. Range: 0 through 4094 This parameter is valid only for family **ccc** mode.
ovlan-priority	Configure the priority value for the IEEE 802.1p bit in the outer VLAN tag. Range: 0 through 7 **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.

Table 107: Parameter for test-name configuration *(continued)*

Parameters	Description
packet-loss-priority	Specify the packet loss priority (PLP) value. If a value is not configured, then the default value of **low** is used. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
reflect-etype	Specify the EtherType ID to be used for reflection of test frames. This parameter is valid only in mode **reflect**. If not specified, then all EtherTypes are reflected. Range: 1 through 65,535. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
reflect-mode	Specify the reflection mode (**mac-rewrite | mac-swap | no-mac-swap**). • **mac-rewrite**—MAC values specified in **source-mac-address** and **destination-mac-address** would be used. • **mac-swap**—Swaps the source-mac-adddress and destination-mac-address in the test frame. This is the default behavior. • **no-mac-swap**—Does not swap MAC addresses. Test frames are returned back as-is.
reflector-port	Port used to configure reflector functionality for RFC 2544 test. The range of ports that can be used based on the front panel port number are: • On ACX5048 [16 through 53] • On ACX5096 [64 through 95, 100 through 103].
service-type	Specify the service type (E-LINE, or E-LAN)
skip-arp-iteration	This parameter is valid only in family **inet** mode. ARP iteration is a 3-second iteration that is run for all **inet** tests. The results of ARP iteration are ignored in test result calculations. The primary use of sending test frames for 3 seconds is to ensure that all devices on the path to destination build their ARP entries. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
source-ipv4-address	Specify the source IPv4 address used for the test frames. If a value is not specified for this parameter, then: • For family **ccc**, if a value is not specified, then by default 192.168.1.10 is used. • For family **inet**, the source address of the interface is used to send out test frames. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.

Table 107: Parameter for test-name configuration *(continued)*

Parameters	Description
source-mac-address	Specify the source MAC address. For example, 0011.2233.4455 This parameter cannot be used when family **inet** is specified. This parameter is optional when family **ccc** is specified. If not specified, then the default value of 0x00:0x60:0x67:0x71:0xC6:0x62 is used.
source-udp-port	Specify the source UDP port number for the test frames. Default: 4040 **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
test-finish-wait-duration	Number of seconds to wait after transmitting the last frame and before concluding that the test as complete. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
test-iterator-duration	Specify the duration of each iteration in seconds. Range: 10 through 120 seconds The default value for test types throughput, back-to-back frames and frame loss rate is 20 seconds. The default value for test type latency is 120 seconds. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
test-interface	Specify the name of the logical interface (UNI) on which the test needs to be run. When you specify the family as **inet** and mode as **initiate-and-terminate** the **test-interface** is ignored, instead the test is run on egress logical interface that is determined by the route lookup on the specified **destination-ipv4-address**. When you specify the family as **inet** and mode as **reflect**, the test-interface is used as the interface to enable reflection service. If test-interface is not specified, a lookup is performed on the **source-ipv4-address** parameter to determine the interface hosting the address. This parameter is mandatory for mode **ethernet-loopback**
test-profile	Specify the name of the **test-profile** to be used for the test. The **test-profile** parameter is ignored when mode **reflect** is used. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.

Table 107: Parameter for test-name configuration *(continued)*

Parameters	Description
vlan-cfi	CFI bit used in the VLAN tag. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
vlan-id	Configure the VLAN ID for the test frames. This parameter is valid only for mode **ethernet-loopback**. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.
vlan-priority	Configure the VLAN priority value. Range: 0 through 7. **NOTE:** This parameter is not supported on ACX5048 and ACX5096 routers.

The following topics describe how to configure a **test-profile** and a **test-name**, start and stop a RFC2544-benchmark test, and copy the test result to a local or a remote file.

Configuring a Test Profile for an RFC 2544-Based Benchmarking Test

You can configure a test profile by including the **test-profile *profile-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level. Table 106 on page 770 lists the parameters for configuring test-profile.

To configure a test profile:

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure an RPM service instance.

```
[edit services]
user@host# edit rpm
```

3. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
```

```
user@host# edit rfc2544-benchmarking
```

4. Define a name for a test profile—for example, profile1.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile profile1
```

5. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1,000 Kbps through 1,000,000 Kbps. Specify a complete decimal number.

```
[edit services rpm rfc2544-benchmarking profiles test-profile profile1]
user@host# set bandwidth-kbps kbps
```

6. Specify the size of the test packet in bytes, with a value from 64 through 9136, to be used for each test iteration. You can specify up to 10 packet sizes, separated by a space, that are used sequentially for the test. The valid packet sizes are 64, 68, 72, 128, 256, 512, 768, 1024, 1280, 1518, 1522, 1600, 1728, 2496, 3584, 4016, 9104, and 9136 bytes. If you specify a packet size other than the ones listed here as valid sizes, the configuration is saved when you commit the setting and no error message is displayed. However, when you start the test by entering the **test services rpm rfc2544-benchmarking test *test-name* start** command, an error message is displayed specifying that you configured an invalid packet size in the test profile associated with the test name.

NOTE:

- The minimum frame size for untagged frames should be 64.
- The minimum frame size for single-tagged frames should be 68.
- The minimum frame size for dual-tagged frames should be 72.

These values are no applicable for **inet**.

```
[edit services rpm rfc2544-benchmarking profiles test-profile profile1]
user@host# set packet-size bytes
```

7. Specify the step percentage for frame-loss tests with a value from 1 through 100. This parameter is not applicable for other test types.

```
[edit services rpm rfc2544-benchmarking profiles test-profile profile1]
```

```
user@host# set step-percent percent-value
```

8. Configure the type of test to be performed.
   - To configure a throughput test, use the **throughput** option with the **test-type** statement.

     ```
     [edit services rpm rfc2544-benchmarking profiles test-profile profile1]
     user@host# set test-type throughput
     ```

   - To configure a latency test, use the **latency** option with the **test-type** statement.

     ```
     [edit services rpm rfc2544-benchmarking profiles test-profile profile1]
     user@host# set test-type latency
     ```

   - To configure a frame-loss test, use the **frame-loss** option with the **test-type** statement.

     ```
     [edit services rpm rfc2544-benchmarking profiles test-profile profile1]
     user@host# set test-type frame-loss
     ```

   - To configure a back-to-back frames test, use the **back-back-frames** option with the **test-type** statement.

     ```
     [edit services rpm rfc2544-benchmarking profiles test-profile profile1]
     user@host# set test-type back-back-frames
     ```

Configuring a Test Name for an RFC 2544-Based Benchmarking Test

You can configure a test name by including the **test-name *test-name*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level.

To configure a test name and define its attributes for initiator:

1. In configuration mode, go to the **[edit services]** hierarchy level.

   ```
   [edit]
   user@host# edit services
   ```

2. Configure an RPM service instance.

   ```
   [edit services]
   ```

```
user@host# edit rpm
```

3. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

4. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

5. Configure the destination IPv4 address for the test packets. This parameter is required only if you configure an IPv4 family inet. This option is not required if you specify circuit cross-connect (CCC) as the family. If you do not configure the destination IPv4 address, the default value of 192.168.1.20 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-ipv4-address address
```

6. Specify the source MAC address used in generated test frames. This parameter is effective for a CCC family and it is not applicable for an **inet** family. If you specify this parameter for an **inet** family, a commit error occurs when you commit the configuration. This parameter is optional for a CCC family. If you do not configure the destination MAC address, the default value of 0x00:0x60:0x67:0x71:0xC6:0x62 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-mac-address address
```

7. Specify the destination MAC address used in generated test frames.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-mac-address address
```

8. Specify the logical interface on which the RFC 2544-based benchmarking test is run. This is a local user-to-network interface (UNI) on behalf of which the test frames are generated when the test direction is egress.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface interface-name
```

9. Specify the family for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family bridge
```

10. Specify the test mode for the packets that are sent during the benchmarking test. The **initiate-and-terminate** option causes the test frames to be initiated from one end and terminated at the same end. The initiation and termination mode requires a reflector to be configured at the peer end to return the test frames from the peer to the originator. The **reflect** option causes the test frames to be reflected on the chosen service (IPv4, Ethernet, or bridge).

    - To configure the initiation and termination mode as the test mode on a router, use the **initiate-and-terminate** option.

    ```
    [edit services rpm rfc2544-benchmarking tests test-name test1]
    user@host# set mode initiate-and-terminate
    ```

    - To configure the reflection mode as the test mode on a router, use the **reflect** option.

    ```
    [edit services rpm rfc2544-benchmarking tests test-name test1]
    user@host# set mode reflect
    ```

11. Specify the direction (**egress | ingress**) of the interface on which the test must be run. The **egress** option causes the test to be run in the egress direction of the interface (traffic sent from user-to-network interface (UNI) toward network-to-network interface (NNI)). The **ingress** option causes the test to be run in the ingress direction of the interface (traffic sent on user-to-network interface (UNI)). You cannot configure **ingress** for a bridge family.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction egress
```

12. Configure the outer VLAN ID for the test frames. This parameter is valid only for a CCC or an Ethernet pseudowire family.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set ovlan-id number
```

13. Configure the inner VLAN ID for the test frames. This parameter is valid only for a CCC or an Ethernet pseudowire family.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set ivlan-id number
```

14. Configure the priority value for the IEEE 802.1p bit in the outer VLAN tag. The priority value is configured when the UNI interface is dual-tagged.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set ovlan-priority value
```

15. Configure the priority value for the IEEE 802.1p bit in the inner VLAN tag. This configuration is optional.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set ivlan-priority value
```

16. Configure the CFI value for the outer VLAN tag. This configuration is optional.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set ovlan-cfi value
```

17. Specify the source IPv4 address to be used in generated test frames. This parameter is optional for both **CCC** and **inet** families. If you do not configure the **source-ipv4-address** for an **inet** family, the source address of the interface is used to transmit the test frames. If you do not configure the **source-ipv4-address** for a **CCC** family, the default value of 192.168.1.10 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-ipv4-address address
```

18. Specify the destination IPv4 address to be used in generated test frames.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-ipv4-address address
```

19. Specify the source UDP port to be used in the UDP header for the generated frames. If you do not specify the UDP port, the default value of 4040 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-udp-port port-number
```

20. Specify the destination UDP port to be used in the UDP header for the generated frames. If you do not specify the UDP port, the default value of 4041 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-udp-port port-number
```

21. Specify the value of the Differentiated Services (DiffServ) field within the IP header of the test frames. The DiffServ code point (DSCP) bits value must be set to a valid 6-bit pattern. If you do not specify this value, 0 is used in the DSCP fields in the IP header.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dscp-code-points dscp-code-bits
```

22. Configure the address type family for the benchmarking test. The **inet** option indicates that the test is run on an IPv4 service. The **ccc** option indicates that the test is run on an CCC or Ethernet pseudowire service. The **direction** statement that you configured in Step 11 specifies the direction (ingress or egress) to be used for the test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

23. Specify the forwarding class to be used for test frames. The forwarding class specifies the manner in which the test frames are processed by the Packet Forwarding Engine of the router. If you do not configure this parameter, test frames are treated as best-effort traffic.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set forwarding-class class-name
```

24. Specify the **halt-on-prefix-down** option to enable a prefix that moves to the down state to cause the corresponding tests to be stopped. The **show** command output for the test displays that the test was aborted because the prefix went down. By default, the RFC 2544-based benchmarking test ignores a prefix-down event (when the prefix associated with the test goes down) and continues to run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set halt-on-prefix-down
```

25. Specify the duration of each iteration in seconds. If you configure this value, the default value of each iteration depends on the type of test being run. For throughput, bursty or back-back-frames, and frame-loss types of tests, the default value is 20 seconds. For latency tests, the default value is 120 seconds.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-iterator-duration seconds
```

26. Specify the name of the test profile to be associated with a particular test name. You must have previously configured the profile by using the **test-profile *profile1*** statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level. The test profile is required when the test mode is configured as initiation and termination. The **test-profile *profile1*** parameter is disregarded when the test mode is configured as reflection. A reflection service does not use the parameters specified in the test profile because the reflection service uses the same parameters for the test frames as the received test frames when it returns the frames to the initiator.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile profile1
```

To configure a test name and define its attributes for reflector:

NOTE: In ACX5048 and ACX5096 routers, while performing RFC 2544 benchmark test, you must ensure that there are no configurations associated with the reflector port.

1. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

2. Configure an RPM service instance.

```
[edit services]
user@host# edit rpm
```

3. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
```

```
user@host# edit rfc2544-benchmarking
```

4. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

5. Specify the test mode for the packets that are sent during the benchmarking test. The **reflect** option causes the test frames to be reflected back to the intiator end..

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

6. Specify the family for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family bridge
```

7. Specify the direction (**egress | ingress**) of the interface on which the test must be run. The **egress** option causes the test to be run in the egress direction of the interface (traffic sent from user-to-network interface (UNI) toward network-to-network interface (NNI)). The **ingress** option causes the test to be run in the ingress direction of the interface (traffic sent on user-to-network interface (UNI)). You cannot configure **ingress** for a bridge family.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction egress
```

8. Configure the destination IPv4 address for the test packets. This parameter is required only if you configure an IPv4 family inet. This option is not required if you specify circuit cross-connect (CCC) as the family. If you do not configure the destination IPv4 address, the default value of 192.168.1.20 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-ipv4-address address
```

9. Specify the source MAC address used in generated test frames. This parameter is effective for a CCC family and it is not applicable for an **inet** family. If you specify this parameter for an **inet** family, a commit

error occurs when you commit the configuration. This parameter is optional for a CCC family. If you do not configure the destination MAC address, the default value of 0x00:0x60:0x67:0x71:0xC6:0x62 is used.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set source-mac-address address
```

10. Specify the destination MAC address used in generated test frames.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set destination-mac-address address
```

11. Specify the logical interface on which the RFC 2544-based benchmarking test is run. This is a local user-to-network interface (UNI) on behalf of which the test frames are generated when the test direction is egress.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface interface-name
```

12. Specify the service type as E-LINE or E-LAN.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set service-type eline | elan
```

13. Specify the forwarding class to be used for test frames. The forwarding class specifies the manner in which the test frames are processed by the Packet Forwarding Engine of the router. If you do not configure this parameter, test frames are treated as best-effort traffic.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set forwarding-class class-name
```

14. Configure the address type family for the benchmarking test. The **inet** option indicates that the test is run on an IPv4 service. The **ccc** option indicates that the test is run on an CCC or Ethernet pseudowire service. The **direction** statement that you configured in Step 7 specifies the direction (ingress or egress) to be used for the test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

15. Specify the EtherType to be used for reflection of the test frames, which is a two-octet field in an Ethernet frame that defines the protocol encapsulated in the frame payload. This parameter is valid only if you configured the test mode to be a reflector. If you do not configure this parameter, all EtherTypes are reflected. Use an EtherType value that matches the EtherType value set on the customer premises equipment (CPE) to which your router connects. The EtherType value appears in the Ethernet type field of the packet. It specifies the protocol being transported in the Ethernet frame. This is an optional parameter.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set reflect-etype ethertype-value
```

16. Specify the reflection mode for the benchmarking test. This configuration is optional.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set reflect-mode (mac-swap | no-mac-swap)
```

You can configure one of the following reflection modes:

- **mac-rewrite**—Enable rewriting of the MAC address on the reflected frames. The MAC addresses specified in the **source-mac-address** and **destination-mac-address** options are used.

  NOTE: In ACX5048 and ACX5096 routers, **mac-rewrite** is not supported in reflection mode.

- **mac-swap**—Swaps the source and destination MAC addresses in the test frame. This is the default behavior.

  NOTE: In ACX5048 and ACX5096 routers, **mac-swap** is not supported in reflection mode.

- **no-mac-swap**—Does not swap the source and destination MAC addresses in the test frame. The frame is returned to the originator without any modification to the MAC addresses.

Starting and Stopping the RFC 2544-Based Benchmarking Test

To start an RFC 2544-based benchmarking test, issue the **run test services rpm rfc2544-benchmarking test *test-name* start** CLI command.

To stop an RFC 2544-based benchmarking test, issue the **run test services rpm rfc2544-benchmarking test *test-name* stop** CLI command.

To start an RFC 2544 benchmarking inet tests on Layer 3 VPN or virtual router, issue the **run test services rpm rfc2544-benchmarking test *test-name* routing-instance *routing-instance-name* start** CLI command.

To stop an RFC 2544 benchmarking inet tests on Layer 3 VPN or virtual router, issue the **run test services rpm rfc2544-benchmarking test *test-name* routing-instance *routing-instance-name* stop** CLI command.

Copying an RFC 2544-Based Benchmarking Test Result

You can copy the RFC 2544-based benchmarking test results to a local or a remote file.

- To copy test results to a local file, use the **run show services rpm rfc2544-benchmarking test-id *number* detail | save rfc-2544-test-result-session-id-*number*** CLI command.
- To copy test results to a remote file, use the **run show services rpm rfc2544-benchmarking test-id *number* detail | save ftp://username:password@sftpchannel.example.com/rfc-2544-test-result-session-id-number**

RELATED DOCUMENTATION

RFC 2544-Based Benchmarking Tests Overview | **763**

Layer 2 and Layer 3 RFC 2544-Based Benchmarking Test Overview | **767**

show services rpm rfc2544-benchmarking | **1371**

show services rpm rfc2544-benchmarking test-id | **1377**

Configuring Ethernet Loopback for RFC 2544-Based Benchmarking Tests

Ethernet loopback is a feature that you can use for verifying the connectivity and identifying or isolating faults in a network.

On ACX Series routers, Ethernet loopback is supported on the egress user-to-network interfaces (UNIs) direction for a **bridge** family configuration. In ACX Series routers, Ethernet loopback is configured on the logical interfaces. The Ethernet loopback feature can be used in performance measurements where packets are looped back to the measuring device for testing various services.

Figure 57: Testing End-to-End Service in Ethernet Loopback Mode

Figure 57 on page 788 shows a scenario where UNI-B interface is configured in Ethernet loopback mode in the egress direction. The packets received on the network-to-network interface (NNI) of the ACX Series router are forwarded to the UNI-B interface and looped back at the UNI-B interface after the source and destination MAC addresses are swapped. This is a use case for testing an end-to-end service.

You can use the following optional parameters to identify an egress traffic flow for Ethernet loopback:

- Source MAC address
- Destination MAC address
- Source IPv4 address
- Destination IPv4 address
- VLAN
- VLAN .1p priority
- EtherType
- Test iterator duration

While performing RFC2544 benchmarking tests, configure Ethernet loopback as the test mode on a logical interface by including the **Ethernet-loopback** CLI statement at the **[edit services rpm rfc2544-benchmarking]** hierarchy level.

If you configure Ethernet loopback on logical interfaces without configuring any of the optional parameters, then any unknown unicast traffic in the same bridge domain also gets looped back and does not get forwarded to other logical interfaces while the test is being performed.

When an RFC2544 benchmarking test is being performed, if the **test-iterator-duration** parameter is not configured, then Ethernet loopback continues until the test is completed or aborted.

> **NOTE:** When performing RFC2544 benchmarking tests, you can configure the test in initiator, reflector, or loopback mode. You cannot perform the RFC2544 benchmarking tests in a combination of these test modes.

The following is a sample Ethernet loopback configuration:

```
[edit services rpm rfc2544-benchmarking]
   tests {
      test-name test1{
         source-mac-address 00:bb:cc:dd:ee:ff;
         destination-mac-address 00:11:22:33:44:55;
         vlan-id 100;
         vlan-priority 2;
         vlan-cfi 1;
         ip-swap;
         udp-tcp-port-swap;
         forwarding-class network-control;
         packet-loss-priority medium-high;
         mode ethernet-loopback;
         family bridge;
         reflect-etype 2048;
         direction egress;
         source-udp-port 2020;
         destination-udp-port 3030;
         test-iterator-duration 50;
         test-interface ge-0/1/6.0;
      }
   }
[edit interfaces]
   ge-0/1/4 {
      flexible-vlan-tagging;
   encapsulation flexible-ethernet-services;
      unit 0 {
         encapsulation vlan-bridge;
         vlan-id 1000;
         family bridge {
            filter {
               input ft1;
            }
         }
      }
   ge-0/1/6 {
      flexible-vlan-tagging;
      encapsulation flexible-ethernet-services;
      unit 0 {
         encapsulation vlan-bridge;
         vlan-id 100;
         input-vlan-map {
            push;
            vlan-id 1000;
```

```
        }
        output-vlan-map pop;
      }
    }
  [edit routing-options]
    ppm {
      traceoptions {
        file ppmd size 100m;
        flag packet;
        flag event;
        flag distribute;
        flag pipe;
        flag all;
      }
    }
  [edit firewall]
    family bridge {
      filter ft1 {
        term t1 {
          from {
            user-vlan-id 100;
          }
          then count loopback;
        }
      }
    }
  [edit bridge-domains]
    bd1 {
      interface ge-0/1/4.0;
      interface ge-0/1/6.0;
    }
```

RELATED DOCUMENTATION

- RFC 2544-Based Benchmarking Tests Overview | **763**
- Layer 2 and Layer 3 RFC 2544-Based Benchmarking Test Overview | **767**
- Configuring RFC 2544-Based Benchmarking Tests | **770**
- RFC 2544-Based Benchmarking Test States | **791**
- Example: Configuring an RFC 2544-Based Benchmarking Test for Layer 3 IPv4 Services | **792**
- Example: Configuring an RFC 2544-Based Benchmarking Test for NNI Direction of Ethernet Pseudowires | **803**

RFC 2544-Based Benchmarking Test States

When you trigger an RFC 2544-based benchmarking test, it passes through a series of states. These states are displayed in the **Test state** field in the brief or detailed output of the **show services rpm rfc2544-benchmarking** command. The following are the names of the states through which the test progresses after it is initiated:

1. RFC2544_TEST_STATE_START_REQUEST—This is the first state that all the triggered tests enter. When a test enters this state, the state denotes that a request has been sent to a Packet Forwarding Engine to start the test.
2. RFC2544_TEST_STATE_START_FAILED—This state indicates that the test failed to start. This state occurs when the Packet Forwarding Engine responds to the **START_REQUEST** message. The Status field of the brief or detailed output of the **show** command displays a reason for the failure. When a test enters this state, it is categorized as an aborted test.
3. RFC2544_TEST_STATE_RUNNING—This state occurs if the Packet Forwarding Engine is able to successfully start the test. This state indicates that the test is in progress. You can use the output of the **show** command to learn additional information about the test progress.
4. RFC2544_TEST_STATE_STOP_REQUEST—A test enters this state when you use the **test services rpm rfc2544-benchmarking *test-id* stop** command. A request is sent to the Packet Forwarding Engine to stop the test.
5. RFC2544_TEST_STATE_STOP_FAILED—This state is entered when the Packet Forwarding Engine failed to stop a test after it received the **STOP_REQUEST** message. The Status field displays further information regarding the exact reason for failure.
6. RFC2544_TEST_STATE_STOPPED—This state is entered when the Packet Forwarding Engine successfully managed to stop a test when it received the **STOP_REQUEST** message.
7. RFC2544_TEST_STATE_COMPLETED—This state is entered when the test successfully completes all necessary test steps.
8. RFC2544_TEST_STATE_ABORTED_TIMEOUT—When a request is sent to the Packet Forwarding Engine for any test, a 10-second timer control is started. If a response is not received from the Packet Forwarding Engine and the timer elapses, the test is transitioned to the **ABORTED_TIMEOUT** state. This state is introduced to prevent a test from indefinitely waiting to receive a reply from the Packet Forwarding Engine.
9. RFC2544_TEST_STATE_RUNTIME_ERROR—This state is entered if the Packet Forwarding Engine encounters an error when the test is running. The Status field of the brief or detailed output specifies the reason for the failure. Tests that encounter the **RUNTIME_ERROR** state are added to the count of

the aborted-tests category, which can be viewed from the output of the **show services rpm rfc2544-benchmarking** command.

RELATED DOCUMENTATION

- RFC 2544-Based Benchmarking Tests Overview | **763**
- Layer 2 and Layer 3 RFC 2544-Based Benchmarking Test Overview | **767**
- Configuring RFC 2544-Based Benchmarking Tests | **770**
- show services rpm rfc2544-benchmarking | **1371**
- show services rpm rfc2544-benchmarking test-id | **1377**

Example: Configuring an RFC 2544-Based Benchmarking Test for Layer 3 IPv4 Services

IN THIS SECTION

This example shows how to configure the benchmarking test for a Layer 3 IPv4 service.

NOTE: This example is not applicable for ACX5048 and ACX5096 routers.

Requirements

This example uses the following hardware and software components:

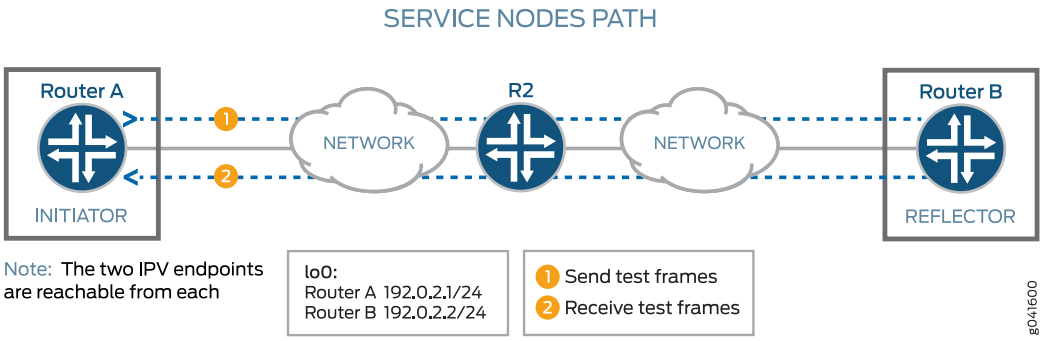
- An ACX Series router
- Junos OS Release 12.3X53 or later

Overview

Consider a sample topology in which a router, Router A, functions as an initiator and terminator of the test frames for an RFC 2544-based benchmarking test. Router A is connected over a Layer 3 network to another router, Router B, which functions as a reflector to reflect back the test frames it receives from Router A. IPv4 is used for transmission of test frames over the Layer 3 network. This benchmarking test is used to compute the IPv4 service parameters between Router A and Router B. Logical interfaces on both the routers are configured with IPv4 addresses to measure the performance attributes, such as throughput, latency, frame loss, and bursty frames, of network devices for the IPv4 service.

Figure 51 on page 668 shows the sample topology to perform an RFC 2544 test for a Layer 3 IPv4 service.

Figure 58: RFC 2544-Based Benchmarking Test for a Layer 3 IPv4 Service



Configuration

IN THIS SECTION

In this example, you configure the benchmarking test for a Layer 3 IPv4 service that is between interface ge-0/0/0 on Router A and interface ge-0/0/4 on Router B to detect and analyze the performance of the interconnecting routers. You need not configure a test profile on Router B because it operates as a reflector.

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:

Configuring Benchmarking Test Parameters on Router A

```
set interfaces ge-0/0/0 unit 0 family inet address 200.0.0.1/24
set rfc2544-benchmarking profiles test-profile throughput test-type throughput
set rfc2544-benchmarking profiles test-profile throughput packet-size 64
set rfc2544-benchmarking profiles test-profile throughput test-duration 20m
set rfc2544-benchmarking profiles test-profile throughput bandwidth-kbps 500
set rfc2544-benchmarking tests test-name test1 test-profile throughput
set rfc2544-benchmarking tests test-name test1 interface ge-0/0/0.1
set rfc2544-benchmarking tests test-name test1 mode initiate-and-terminate
set rfc2544-benchmarking tests test-name test1 family inet
set rfc2544-benchmarking tests test-name test1 dest-address 200.0.0.2
set rfc2544-benchmarking tests test-name test1 udp-port 4001
```

Configuring Benchmarking Test Parameters on Router B

```
set interfaces ge-0/0/4 unit 0 family inet address 200.0.0.2/24
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/4.1
set services rpm rfc2544-benchmarking tests test-name test1 mode reflect
set services rpm rfc2544-benchmarking tests test-name test1 family inet
set services rpm rfc2544-benchmarking tests test-name test1 dest-address 200.0.0.1
set services rpm rfc2544-benchmarking tests test-name test1 udp-port 4001
```

Configuring Benchmarking Test Parameters on Router A

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router A:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/0
```

3. Configure a logical unit and specify the protocol family.

```
[edit interfaces ge-0/0/0]
user@host# edit unit 0 family inet
```

4. Specify the address for the logical interface.

```
[edit interfaces ge-0/0/0 unit 0 family inet]
user@host# set address 200.0.0.1/24
```

5. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit interfaces ge-0/0/0 unit 0 family inet]
user@host# up
```

6. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# top
```

7. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

8. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

9. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

10. Define a name for a test profile—for example, throughput.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile throughput
```

11. Configure the type of test to be performed as throughput.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type throughput
```

12. Specify the size of the test packet as 64 bytes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type packet-size 64
```

13. Specify the period—for example, 20 minutes—for which the test is to be performed in hours, minutes, or seconds by specifying a number followed by the letter h (for hours), m (for minutes), or s (for seconds), respectively.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type test-duration 20m
```

14. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1,000 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type bandwidth-kbps 500
```

15. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# up
```

16. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# up
```

17. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

18. Specify the name of the test profile—for example, throughput—to be associated with a particular test name.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile throughput
```

19. Specify the logical interface, ge-0/0/0.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/0.1
```

20. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode initiate-and-terminate
```

21. Configure the address type family, **inet**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

22. Configure the destination IPv4 address for the test packets.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dest-address 200.0.0.2
```

23. Specify the UDP port of the destination to be used in the UDP header for the generated frames as 4001.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set udp-port 4001
```

24. Start the benchmarking test on the initiator.

```
user@host> test services rpm rfc2544-benchmarking test test1 start
```

After the test is successfully completed, it is automatically stopped at the initiator.

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router B:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/4
```

3. Configure a logical unit and specify the protocol family as **inet**.

```
[edit interfaces ge-0/0/4]
user@host# edit unit 0 family inet
```

4. Specify the address for the logical interface.

```
[edit interfaces ge-0/0/4 unit 0 family inet]
user@host# set address 200.0.0.2/24
```

5. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit interfaces ge-0/0/4 unit 0 family inet]
user@host# up
```

6. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# top
```

7. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

8. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

9. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

10. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

11. Specify the logical interface, ge-0/0/4.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/4.1
```

12. Specify **reflect** as the test mode for the packets that are sent during the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

13. Configure the address type family, **inet**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

14. Configure the destination IPv4 address for the test packets as 200.0.0.1.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dest-address 200.0.0.1
```

15. Specify the UDP port of the destination to be used in the UDP header for the generated frames as 4001.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set udp-port 4001
```

16. Start the benchmarking test on the reflector.

```
user@host> test services rpm rfc2544-benchmarking test test1 start
```

After the test is successfully completed at the initiator, you can stop the test at the reflector by entering the **test services rpm rfc2544-benchmarking test test1 start** command.

Results

In configuration mode, confirm your configuration on Router A and Router B by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Configuring Benchmarking Test Parameters on Router A:

```
[edit interfaces]
  ge-0/0/0 {
      unit 0 {
          family inet {
```

```
            address 200.0.0.1/24;
        }
     }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      profiles {
          test-profile throughput {
              test-type throughput
              packet-size 64;
              test-duration 20m;
              bandwidth-kbps 500;
          }
      }

      tests {
          test-name test1 {
              test-profile throughput;
              interface ge-0/0/0.1;
              mode initiate,terminate;
              family inet;
              dest-address 200.0.0.2
              udp-port 4001;
          }
      }
  }
```

Configuring Benchmarking Test Parameters on Router B:

```
[edit interfaces]
  ge-0/0/4 {
      unit 0 {
          family inet {
              address 200.0.0.2/24;
          }
     }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
```

```
    tests {
        test-name test1 {
            interface ge-0/0/4.1;
            mode reflect;
            family inet;
            dest-address 200.0.0.1;
            udp-port 4001;
        }
    }
}
```

After you have configured the device, enter the **commit** command in configuration mode.

Verifying the Results of the Benchmarking Test for Layer 3 IPv4 Services

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between Router A and Router B.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between Router A and Router B.

Action

In operational mode, enter the **run show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show ptp clock** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

- Configuring RFC 2544-Based Benchmarking Tests | 770
- rfc2544-benchmarking | 1108
- profiles | 1098
- tests | 1177
- show services rpm rfc2544-benchmarking | 1371
- show services rpm rfc2544-benchmarking test-id | 1377

Example: Configuring an RFC 2544-Based Benchmarking Test for NNI Direction of Ethernet Pseudowires

IN THIS SECTION

This example shows how to configure the benchmarking test for a network-to-network interface (NNI) direction of an Ethernet pseudowire service.

Requirements

This example uses the following hardware and software components:

- An ACX Series router
- Junos OS Release 12.3X52 or later

Overview

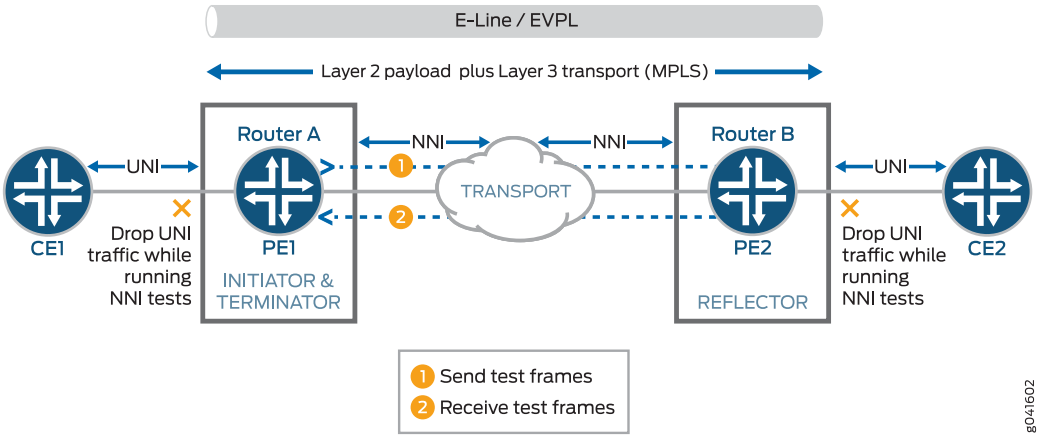
Consider a sample topology in which a router, Router A, functions as an initiator and terminator of the test frames for an RFC 2544-based benchmarking test. Router A operates as a provider edge device, PE1, which is connected to a customer edge device, CE1, on one side and over an Ethernet pseudowire to another router, Router B, which functions as a reflector to reflect back the test frames it receives from

Router A. Router B operates as a provider edge device, PE2, which is the remote router located at the other side of the service provider core. The UNI direction of CE1 is connected to the NNI direction of PE1. An MPLS tunnel connects PE1 and PE2 over the Ethernet pseudowire or the Ethernet line (E-LINE).

This benchmarking test is used to compute the performance attributes in the network-to-network interface (NNI) direction of an Ethernet pseudowire service between Router A and Router B. The logical interface under test on Router A is the CE1 interface with UNI as the direction, and the logical interface under test on Router B is the CE2 interface with NNI as the direction. Data traffic arriving from UNI towards NNI is ignored while the test is in progress. Packets from NNI are not sent toward the customer edge because all packets are assumed to be test frames. The CCC family and NNI direction are configured on routers A and B.

Figure 53 on page 691 shows the sample topology to perform an RFC 2544 test for the NNI direction of an Ethernet pseudowire service.

Figure 59: RFC 2544-Based Benchmarking Test for NNI Direction of an Ethernet Pseudowire



Configuration

IN THIS SECTION

In this example, you configure the benchmarking test for the NNI direction of an Ethernet pseudowire service that is enabled between two routers to detect and analyze the performance of the interconnecting routers.

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:

Configuring Benchmarking Test Parameters on Router A

```
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 0 encapsulation vlan-ccc
set interfaces ge-0/0/0 unit 0 vlan-id 101
set services rpm rfc2544-benchmarking profiles test-profile throughput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile throughput packet-size 64
set services rpm rfc2544-benchmarking profiles test-profile throughput test-duration 20
set services rpm rfc2544-benchmarking profiles test-profile throughput bandwidth-kbps 500
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/0.1
set services rpm rfc2544-benchmarking tests test-name test1 test-profile throughput
set services rpm rfc2544-benchmarking tests test-name test1 mode initiate,terminate
set services rpm rfc2544-benchmarking tests test-name test1 family ccc
set services rpm rfc2544-benchmarking tests test-name test1 direction nni
```

Configuring Benchmarking Test Parameters on Router B

```
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4 unit 0 encapsulation vlan-ccc
set interfaces ge-0/0/4 unit 0 vlan-id 101
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/4.1
set services rpm rfc2544-benchmarking tests test-name test1 mode reflect
set services rpm rfc2544-benchmarking tests test-name test1 reflector-port 25
set services rpm rfc2544-benchmarking tests test-name test1 mode family ccc
set services rpm rfc2544-benchmarking tests test-name test1 direction uni
```

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following example 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 test parameters on Router A:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/0
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/0]
user@host# set vlan-tagging
```

4. Configure a logical unit for the interface.

```
[edit interfaces ge-0/0/0]
user@host# edit unit 0
```

5. Specify the encapsulation for Ethernet VLAN circuits.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# set encapsulation vlan-ccc
```

6. Configure the VLAN ID on the logical interface.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for a test profile—for example, throughput.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile throughput
```

12. Configure the type of test to be performed as throughput.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type throughput
```

13. Specify the size of the test packet as 64 bytes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type packet-size 64
```

14. Specify the period—for example, 20 minutes—for which the test is to be performed in hours, minutes, or seconds by specifying a number followed by the letter h (for hours), m (for minutes), or s (for seconds).

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type test-duration 20m
```

15. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type bandwidth-kbps 500
```

16. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# up
```

17. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# up
```

18. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

19. Specify the name of the test profile—for example, throughput—to be associated with a particular test name.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile throughput
```

20. Specify the logical interface, ge-0/0/0.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/0.1
```

21. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode initiate-and-terminate
```

22. Configure the address type family, **ccc**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

23. Specify the direction of the interface on which the test must be run, which is NNI in this example.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction nni
```

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following example 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 test parameters on Router B:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/4
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/4]
user@host# set vlan-tagging
```

4. Configure a logical unit for the interface.

```
[edit interfaces ge-0/0/4]
user@host# edit unit 0
```

5. Specify the encapsulation for Ethernet VLAN circuits.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set encapsulation vlan-ccc
```

6. Configure the VLAN ID on the logical interface.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

12. Specify the logical interface, ge-0/0/4.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/4.1
```

13. Specify **reflect** as the test mode for the packets that are sent during the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

14. Configure the address type family, **ccc**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

15. Specify the direction of the interface on which the test must be run, which is NNI in this example.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction nni
```

Results

In configuration mode, confirm your configuration on Router A and Router B by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Configuring Benchmarking Test Parameters on Router A:

```
[edit interfaces]
  ge-0/0/0 {
      vlan-tagging;
      unit 0 {
          encapsulation vlan-ccc;
          vlan-id 101;
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      profiles {
          test-profile throughput {
```

```
            test-type throughput
            packet-size 64;
            test-duration 20m;
            bandwidth-kbps 500;
        }
    }

    tests {
        test-name test1 {
            interface ge-0/0/0.1;
            test-profile throughput;
            mode initiate,terminate;
            family ccc;
            direction nni;
        }
    }
}
```

Configuring Benchmarking Test Parameters on Router B:

```
[edit interfaces]
  ge-0/0/4 {
      vlan-tagging;
      unit 0 {
          encapsulation vlan-ccc;
          vlan-id 101;
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
      tests {
          test-name test1 {
              interface ge-0/0/4.1;
              mode reflect;
              family ccc;
              direction nni;
          }
      }
  }
```

After you have configured the device, enter the **commit** command in configuration mode.

Verifying the Results of the Benchmarking Test for NNI Direction of an Ethernet Pseudowire Service

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between Router A and Router B.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between Router A and Router B.

Action

In operational mode, enter the **run show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

- Configuring RFC 2544-Based Benchmarking Tests | **770**
- rfc2544-benchmarking | **1108**
- profiles | **1098**
- tests | **1177**
- show services rpm rfc2544-benchmarking | **1371**
- show services rpm rfc2544-benchmarking test-id | **1377**

Example: Configuring an RFC 2544-Based Benchmarking Test for UNI Direction of Ethernet Pseudowires

IN THIS SECTION

This example shows how to configure the benchmarking test for the user-to-network interface (UNI) direction of an Ethernet pseudowire service.

Requirements

This example uses the following hardware and software components:

- An ACX Series router
- Junos OS Release 12.3X53 or later

Overview

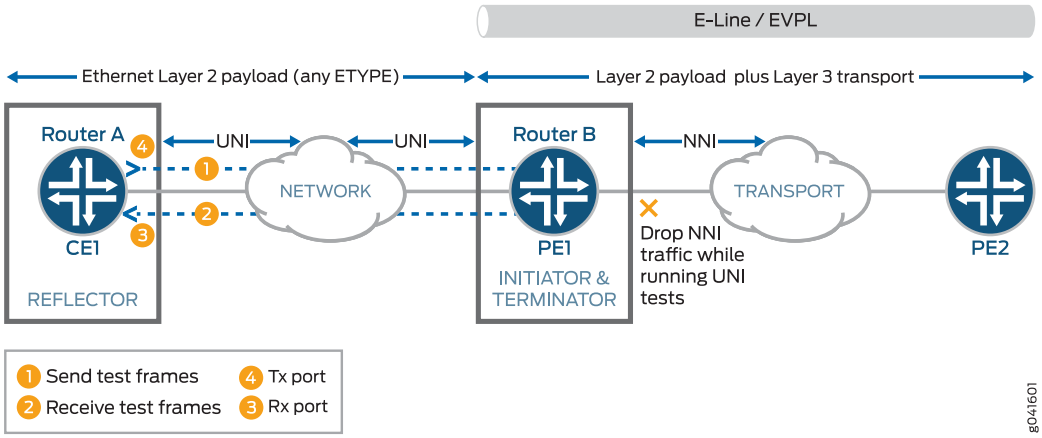
Consider a sample topology in which a router, Router A, functions as a reflector of the test frames for an RFC 2544-based benchmarking test. The logical customer edge (CE)-facing interface and **inet** family are configured on Router A. Router A is not part of a pseudowire and therefore, a Layer 3 family configuration is required on it. Router A, which is a customer edge device, CE1, is connected to Router B, which functions as a provider edge device, PE1, over an Ethernet pseudowire in the UNI direction with EtherType or Layer 2 Ethernet payload. The logical interface, CCC family, and UNI direction are configured on Router B. Router B or PE1 is connected over an Ethernet pseudowire in the NNI direction to a provider edge device at the remote site, PE2. The link between CE1 and PE1 is an Ethernet Layer 2 network and it can be configured with any EtherType value. The link between PE1 and PE2 is an Ethernet line (E-LINE) or an Ethernet Private Line (EPL) that has Layer 2 payload and Layer 3 transport sent over it. Router B or PE1 functions as an initiator and terminator of the test frames that are sent to Router A and reflected back from it.

This benchmarking test is used to compute the performance attributes in the user-to-network interface (UNI) direction of an Ethernet pseudowire service between Router A and Router B. Data traffic arriving from a network-to-network interface (NNI) toward the customer edge is ignored while the test is in

progress. Packets from the CE are not sent toward the NNI because all packets are assumed to be test probes.

Figure 52 on page 679 shows the sample topology to perform an RFC 2544 test for the UNI direction of an Ethernet pseudowire service.

Figure 60: RFC 2544-Based Benchmarking Test for UNI Direction of an Ethernet Pseudowire



Configuration

IN THIS SECTION

In this example, you configure the benchmarking test for the UNI direction of an Ethernet pseudowire service that is enabled between two routers to detect and analyze the performance of the interconnecting routers.

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:

Configuring Benchmarking Test Parameters on Router A

```
set interfaces ge-0/0/0 vlan-tagging
set interfaces ge-0/0/0 unit 0 vlan-id 101
set interfaces ge-0/0/0 unit 0 family inet address 200.0.0.1/24
set services rpm rfc2544-benchmarking profiles test-profile throughput test-type throughput
set services rpm rfc2544-benchmarking profiles test-profile throughput packet-size 64
set services rpm rfc2544-benchmarking profiles test-profile throughput test-duration 20m
set services rpm rfc2544-benchmarking profiles test-profile throughput bandwidth-kbps 500
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/0.1
set services rpm rfc2544-benchmarking tests test-name test1 test-profile throughput
set services rpm rfc2544-benchmarking tests test-name test1 mode initiate,terminate
set services rpm rfc2544-benchmarking tests test-name test1 family inet
set services rpm rfc2544-benchmarking tests test-name test1 dest-address 200.0.0.2
set services rpm rfc2544-benchmarking tests test-name test1 udp-port 4001
```

Configuring Benchmarking Test Parameters on Router B

```
set interfaces ge-0/0/4 vlan-tagging
set interfaces ge-0/0/4 unit 0 encapsulation vlan-ccc
set interfaces ge-0/0/4 unit 0 vlan-id 101
set services rpm rfc2544-benchmarking tests test-name test1 interface ge-0/0/4.1
set services rpm rfc2544-benchmarking tests test-name test1 mode reflect
set services rpm rfc2544-benchmarking tests test-name test1 mode family ccc
set services rpm rfc2544-benchmarking tests test-name test1 direction uni
```

Configuring Benchmarking Test Parameters on Router A

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router A:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
```

```
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/0
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/0]
user@host# set vlan-tagging
```

4. Configure a logical unit and specify the protocol family as **inet**.

```
[edit interfaces ge-0/0/0]
user@host# edit unit 0 family inet
```

5. Specify the address for the logical interface.

```
[edit interfaces ge-0/0/0 unit 0 family inet]
user@host# set address 200.0.0.1/24
```

6. Configure the VLAN ID on the logical interface as 101.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/0 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for a test profile—for example, throughput.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit profiles test-profile throughput
```

12. Configure the type of test to be performed as throughput.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type throughput
```

13. Specify the size of the test packet as 64 bytes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type packet-size 64
```

14. Specify the period for which the test is to be performed in hours, minutes, or seconds by specifying a number followed by the letter h (for hours), m (for minutes), or s (for seconds). In this example, you configure the period as 20 minutes.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type test-duration 20m
```

15. Define the theoretical maximum bandwidth for the test in kilobits per second, with a value from 1 Kbps through 1,000,000 Kbps.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# set test-type bandwidth-kbps 500
```

16. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles test-profile throughput]
user@host# up
```

17. Enter the **up** command to go the previous level in the configuration hierarchy.

```
[edit services rpm rfc2544-benchmarking profiles]
user@host# up
```

18. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

19. Specify the name of the test profile—for example, throughput—to be associated with a particular test name.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-profile throughput
```

20. Specify the logical interface, ge-0/0/0.1, on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/0.1
```

21. Specify the test mode for the packets that are sent during the benchmarking test as initiation and termination.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode initiate-and-terminate
```

22. Configure the address type family, **inet**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family inet
```

23. Configure the destination IPv4 address for the test packets as 200.0.0.2.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set dest-address 200.0.0.2
```

24. Specify the UDP port of the destination to be used in the UDP header for the generated frames as 4001.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set udp-port 4001
```

Configuring Benchmarking Test Parameters on Router B

Step-by-Step Procedure

The following example requires you to navigate various levels in the configuration hierarchy. For information about navigating the CLI, see *Using the CLI Editor in Configuration Mode* in the *CLI User Guide*.

To configure the test parameters on Router B:

1. In configuration mode, go to the **[edit interfaces]** hierarchy level:

```
[edit]
user@host# edit interfaces
```

2. Configure the interface on which the test must be run.

```
[edit interfaces]
user@host# edit ge-0/0/4
```

3. Configure VLAN tagging for transmission and reception of 802.1Q VLAN-tagged frames.

```
[edit interfaces ge-0/0/4]
user@host# set vlan-tagging
```

4. Configure a logical unit for the interface.

```
[edit interfaces ge-0/0/4]
user@host# edit unit 0
```

5. Specify the encapsulation for Ethernet VLAN circuits.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set encapsulation vlan-ccc
```

6. Configure the VLAN ID as 101 on the logical interface.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# set vlan-id 101
```

7. Go to the top level of the configuration command mode.

```
[edit interfaces ge-0/0/4 unit 0]
user@host# top
```

8. In configuration mode, go to the **[edit services]** hierarchy level.

```
[edit]
user@host# edit services
```

9. Configure a real-time performance monitoring service (RPM) instance.

```
[edit services]
user@host# edit rpm
```

10. Configure an RFC 2544-based benchmarking test for the RPM instance.

```
[edit services rpm]
user@host# edit rfc2544-benchmarking
```

11. Define a name for the test—for example, test1. The test name identifier can be up to 32 characters in length.

```
[edit services rpm rfc2544-benchmarking]
user@host# edit tests test-name test1
```

12. Specify the logical interface on which the RFC 2544-based benchmarking test is run.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set test-interface ge-0/0/4.1
```

13. Specify **reflect** as the test mode for the packets that are sent during the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set mode reflect
```

14. Configure the address type family, **ccc**, for the benchmarking test.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set family ccc
```

15. Specify the direction of the interface on which the test must be run, which is UNI in this example.

```
[edit services rpm rfc2544-benchmarking tests test-name test1]
user@host# set direction uni
```

Results

In configuration mode, confirm your configuration on Router A and Router B by entering the **show** command. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Configuring Benchmarking Test Parameters on Router A:

```
[edit interfaces]
  ge-0/0/0 {
      vlan-tagging;
      unit 0 {
          vlan-id 101;
          family inet {
              address 200.0.0.1/24;
          }
      }
  }

  [edit services rpm]
```

```
  rfc2544-benchmarking {
      profiles {
          test-profile throughput {
              test-type throughput
              packet-size 64;
              test-duration 20m;
              bandwidth-kbps 500;
          }
      }

      tests {
          test-name test1 {
              interface ge-0/0/0.1;
              test-profile throughput;
              mode initiate,terminate;
              family inet;
              dest-address 200.0.0.2
              udp-port 4001;
          }
      }
  }
```

Configuring Benchmarking Test Parameters on Router B:

```
[edit interfaces]
  ge-0/0/4 {
      vlan-tagging;
      unit 0 {
          encapsulation vlan-ccc;
          vlan-id 101;
      }
  }

  [edit services rpm]
  rfc2544-benchmarking {
      # Note, When in reflector mode, test profile is not needed
      tests {
          test-name test1 {
              interface ge-0/0/4.1;
              mode reflect;
              family ccc;
              direction uni;
          }
```

```
    }
}
```

After you have configured the device, enter the **commit** command in configuration mode.

Verifying the Results of the Benchmarking Test for UNI Direction of an Ethernet Pseudowire Service

IN THIS SECTION

Examine the results of the benchmarking test that is performed on the configured service between Router A and Router B.

Verifying the Benchmarking Test Results

Purpose

Verify that the necessary and desired statistical values are displayed for the benchmarking test that is run on the configured service between Router A and Router B.

Action

In operational mode, enter the **run show services rpm rfc2544-benchmarking (aborted-tests | active-tests | completed-tests | summary)** command to display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance.

Meaning

The output displays the details of the benchmarking test that was performed. For more information about the **run show services rpm rfc2544-benchmarking** operational command, see **show services rpm rfc2544-benchmarking** in the CLI Explorer.

RELATED DOCUMENTATION

Configuring RFC 2544-Based Benchmarking Tests | **770**

rfc2544-benchmarking | 1108
profiles | 1098
tests | 1177
show services rpm rfc2544-benchmarking | 1371
show services rpm rfc2544-benchmarking test-id | 1377

Configuring a Service Package to be Used in Conjunction with PTP

On ACX1100 routers, you can configure a service package on the router for the RFC 2544-based benchmarking test, or for NAT and IPsec applications. When you configure the service package for the RFC 2544-based benchmarking test or for the NAT and IPsec applications, a reboot of the Forwarding Engine Board (FEB) occurs to apply the service package selection. By default, the service package for RFC 2544 benchmarking test is selected. The selection of a service package is needed on ACX1100 routers when you configure such routers for IEEE 1588v2 Precision Time Protocol (PTP) because both RFC 2544-based benchmarking tests and a combination of NAT and IPsec protocols are not supported simultaneously; you can configure only PTP and RFC 2544-based tests, or PTP and the combination of NAT and IPsec at a point in time.

You need to specify the service package to be RFC 2544-based or NAT and IPsec-based only for ACX1100-AC routers. The selection of a service package is not needed on ACX Series routers other than the ACX1100-AC and ACX500 routers because on such routers, only the RFC 2544-based benchmarking tests are supported; NAT and IPsec applications are not supported on those routers.

To configure the RFC 2544-based service package on a particular FPC, include the **service-package bundle-rfc2544** statement at the **[edit chassis fpc *slot-number*]** hierarchy level.

```
[edit chassis]
fpc slot-number {
    service-package bundle-rfc2544;
}
```

To configure the NAT and IPsec applications service package on a particular FPC, include the **service-package bundle-nat-ipsec** statement at the **[edit chassis fpc *slot-number*]** hierarchy level.

```
[edit chassis]
fpc slot-number {
    service-package bundle-nat-ipsec;
}
```

RELATED DOCUMENTATION

service-package

CHAPTER 18

Tracking Streaming Media Traffic Using Inline Video Monitoring

IN THIS CHAPTER

Understanding Inline Video Monitoring on MX Series Routers

Junos OS supports inline video monitoring using media delivery index (MDI) metrics.

Before you use the inline video monitoring feature, ensure that you understand the following terms:

- **media delivery index**—MDI metrics facilitate identification of buffering needs for streaming media. Buffering must be adequate to compensate for packet jitter, measured by the MDI delay factor, and quality problems indicated by lost packets, measured by the MDI media loss rate (MLR). By performing measurements under varying load conditions, you can identify sources of significant jitter or packet loss and take appropriate action.
- **delay factor**—Delay factor is the maximum observed time difference between the arrival of media data and the drain of media data. The expected drain rate is the nominal, constant traffic rate for constant bit rate streams or the computed traffic rate of variable rate media stream packet data.

  For typical stream rates of 1 megabit per second and higher, an interval of one second provides an adequate sample time. The delay factor indicates how long a data stream must be buffered (delayed) at its nominal bit rate to prevent packet loss.

  The delay factor suggests the minimum size of the buffer required at the next downstream node. As a stream progresses, the variation of the delay factor indicates packet bunching or packet gaps (jitter). Greater delay factor values also indicate that more network latency is needed to deliver a stream due to the need to pre-fill a receive buffer before beginning the drain to guarantee no underflow.

When the nominal drain bit rate at a receiving node is known, the delay factor's maximum indicates the size of buffer required to accommodate packet jitter.

- **Media rate variation (MRV)**—This value is the difference between the expected packet rate and actual packet rate, expressed as a percentage of the expected packet rate.
- **Media loss rate (MLR)**—This value is the number of media packets lost over a configurable time interval (***interval-duration***), where the flow packets carry streaming application information. A single IP packet can contain zero or more streaming packets. For example, an IP packet typically contains seven 188-byte MPEG transport stream packets. In this case, a single IP packet loss results in seven lost packets counted (if those seven lost packets did not include null packets). Including out-of-order packets is important, because many consumer-type streaming devices do not attempt to reorder packets that are received out of order.

To configure the monitoring process, define criteria templates and apply them to the interfaces and flows you want to monitor. Monitoring templates include the following criteria:

- Duration of each measurement cycle
- Flow rate information used to establish expected flow rates
- Threshold levels for media rate variation and media loss rate that trigger desired syslog alerts

For each interface you want to monitor, you can define one or more filters to select IPv4 flows for monitoring. Flows are designated as input or output flows. Starting in Junos OS Release 17.2R1, you can identify IPv4-over-MPLS flows. Starting in Junos OS Release 17.4R1, you can identify IPv6 flows and IPv6-over MPLS flows. Starting in Junos OS Release 19.1R1, you can configure MX-Series Routers for inline video monitoring of uncompressed HD or 4K stream video (Payload Type 98 and 99). MDI functionality has been extended to video flows such as ST 2000-5 (RTP PT 98) and ST 2000-6 (RTP PT 99). These are non-MPEG video flows over IP/UDP/RTP and are constant bit-rate flows. The operator would specify proper IP addresses and UDP ports so that non-video flows over RTP will not go through MDI processing.

MPLS flows with more than three labels cannot be monitored.

IPv4 flows are uniquely identified by:

- Destination IP address
- Destination port
- Source IP address
- Source port
- Direction
- Interface index
- Media type (RTP or MPEG)

IPv4-over-MPLS flows are uniquely identified by:

- The top three MPLS labels
- Destination IP address
- Destination port
- Source IP address
- Source port
- Direction
- Interface index
- Media type (RTP or MPEG)

IPv6 flows are uniquely identified by:

- Destination IP address
- Destination port
- Source IP address
- Source port
- Direction
- Interface index
- Media type (RTP or MPEG)

IPv6-over-MPLS flows are uniquely identified by:

- The top three MPLS labels
- Destination IP address
- Destination port
- Source IP address
- Source port
- Direction
- Interface index
- Media type (RTP or MPEG)

Junos OS supports the definition of filters for up to 256 flows on an interface, which can consist of input flows, output flows, or a combination of input and output flows. These filters provide criteria for selecting flows for monitoring. If the selection criteria consist of lists of IP addresses or ports, you can exceed the maximum number of match conditions for flows. Video monitoring selects a widely variable number of flows based on flow filters.

The total number of destination IP addresses configured in a flow for an interface cannot exceed 32, and the total number of source IP addresses configured in a flow for an interface cannot exceed 32.

Inline video monitoring is not supported when you enable Next Gen Services on an MX Series router.

Inline video monitoring is available on MX Series 5G Universal Routing Platforms using only the following MPCs:

- MPC1
- MPC1E
- MPC2
- MPC2E
- MPC2E-NG
- MPC3E-NG
- MPC-16XGE
- MPC5E
- MPC6E
- MPC7E
- MPC8E
- MPC9E

> **NOTE:** Traffic throughput is reduced below the interface bandwidth when video monitoring is used with an MPC2E-NG or MPC3E-NG in the following scenario:
>
> - The input and output ports are on the same slot.
> - The input-flows is configured as inet and the output-flows is configured as mpls.
> - At least one flow has a traffic rate greater than 2 Gbps.
>
> To avoid this reduced throughput, use input and output ports on different slots.

Starting in Junos OS Release 16.1R1, you can configure the number of flows that can be measured per Packet Forwarding Engine at a time, up to a value of 8192. The maximum configured number of flows that can be measured for each MPC model is shown in the second column of Table 108 on page 831. The default number of flows that can be measured for each MPC model is shown in the third column of Table 108 on page 831. In Junos OS Release 15.1 and earlier, you cannot configure the number of flows that can be measured.

When you do not define input or output flow filters for a monitored interface, all flows on the interface are subject to monitoring.

Table 108: MPC Flow Monitoring Capacity by Model

MPC Model	Maximum Configurable Number of Flows Monitored Simultaneously (Starting in Junos OS Release 16.1)	Default Number of Flows Monitored Simultaneously
MPC1	8000	1000
MPC1E	8000	1000
MPC2	16,000	2000
MPC2E	16,000	2000
MPC2E-NG	8000	1000
MPC3E-NG	8000	1000
MPC-16XGE	32,000	4000
MPC5E	40,000	5000
MPC6E	40,000	5000
MPC7E	40,000	5000
MPC8E	40,000	5000
MPC9E	40,000	5000

NOTE: Junos OS measures both UDP flows (the default) and RTP flows. Junos OS differentiates media traffic over UDP or RTP by inspecting the first byte in the UDP payload. If the first byte of the UDP payload is 0x47 (MPEG2-TS sync byte), the traffic is treated as media traffic over UDP. Traffic is treated as media traffic over RTP if the version field is 2 and the payload type is 33 in the RTP header. When neither of these criteria are met, the packet is not considered for video monitoring.

Starting in Junos OS Release 15.1R1, MX Series routers support the inline video monitoring to measure media delivery index (MDI) metrics that can be accessed using the SNMP GET operation. Currently, inline

MDI can generate only a system log when the computed value is not within the configured range. SNMP is primarily used to monitor alarms raised by the inline video monitoring feature. The alarms are monitored in the network management systems either to troubleshoot the problem or to diagnose degradation in video quality.

You use the **video-monitoring** statement at the **[edit services]** hierarchy level to specify monitoring criteria for two key indicators of video traffic problems: delay factor and media loss rate (MLR), and to apply these metrics to flows on designated interfaces.

Release History Table

Release	Description
19.3R2	Inline video monitoring is not supported when you enable Next Gen Services on an MX Series router.
19.1R1	Starting in Junos OS Release 19.1R1, you can configure MX-Series Routers for inline video monitoring of uncompressed HD or 4K stream video (Payload Type 98 and 99). MDI functionality has been extended to video flows such as ST 2000-5 (RTP PT 98) and ST 2000-6 (RTP PT 99). These are non-MPEG video flows over IP/UDP/RTP and are constant bit-rate flows. The operator would specify proper IP addresses and UDP ports so that non-video flows over RTP will not go through MDI processing.
17.4R1	Starting in Junos OS Release 17.4R1, you can identify IPv6 flows and IPv6-over MPLS flows.
17.2R1	Starting in Junos OS Release 17.2R1, you can identify IPv4-over-MPLS flows.
16.1R1	Starting in Junos OS Release 16.1R1, you can configure the number of flows that can be measured per Packet Forwarding Engine at a time, up to a value of 8192.
15.1R1	Starting in Junos OS Release 15.1R1, MX Series routers support the inline video monitoring to measure media delivery index (MDI) metrics that can be accessed using the SNMP GET operation. Currently, inline MDI can generate only a system log when the computed value is not within the configured range.

RELATED DOCUMENTATION

Configuring Inline Video Monitoring on MX Series Routers | **833**

show services video-monitoring mdi stats fpc-slot | **1437**

show services video-monitoring mdi errors fpc-slot | **1428**

show services video-monitoring mdi flows fpc-slot | **1430**

alarms | **869**

Configuring Inline Video Monitoring on MX Series Routers

IN THIS SECTION

To configure inline video monitoring, perform the following tasks:

Configuring Media Delivery Indexing Criteria

To configure media delivery indexing criteria:

1. In edit mode, create a named template for video monitoring.

   ```
   user@host# edit services video-monitoring templates template-name
   ```

   For example,

   ```
   user@host# edit services video-monitoring templates t1
   ```

2. Set the duration for sampling in seconds. Flow media delivery indexing statistics are updated at the end of this interval.

   ```
   [edit services video-monitoring templates template-name]
   user@host# set interval-duration interval-duration
   ```

   For example,

   ```
   [edit services video-monitoring templates t1]
   user@host# set interval-duration 1
   ```

BEST PRACTICE: If you change the interval duration when a template is being used, you cause a change in the calculated number of expected packets in an measurement interval for the template. We recommend that you do not change the interval duration for a template that is in use.

3. Set the inactivity timeout.

```
[edit services video-monitoring templates template-name]
user@host# set inactive-timeout inactive-timeout
```

For example,

```
[edit services video-monitoring templates t1]
user@host# set inactive-timeout 30
```

4. Configure either the media rate or layer 3 packet rate to establish expected flow rates used to compare to monitored flow rates.

NOTE: The media rate is the configured media bit rate for the stream. The media rate is used to establish *expected packets per second* (pps).

The Layer 3 packet rate in packets per second (pps) is used to establish *expected bits per second* (bps).

```
[edit services video-monitoring templates template-name]
user@host# set rate media media-bits-per-second
```

For example,

```
[edit services video-monitoring templates t1]
user@host# set rate media 2972400
```

5. Set delay factor thresholds for syslog message levels.

```
[edit services video-monitoring templates template-name]
user@host# set delay-factor threshold info delay-factor-threshold
```

```
user@host# set delay-factor threshold warning delay-factor-threshold
user@host# set delay-factor threshold critical delay-factor-threshold
```

For example,

```
[edit services video-monitoring templates t1]
user@host# set delay-factor threshold info 100
user@host# set delay-factor threshold warning 200
user@host# set delay-factor threshold critical 300
```

6. Set media loss rate thresholds for syslog message levels. You can set the threshold based on number of packets lost, or percentage of packets lost.

```
[edit services video-monitoring templates template-name]
user@host# set media-loss-rate threshold info percentage mlr-percentage
user@host# set media-loss-rate threshold warning percentage mlr-percentage
user@host# set media-loss-rate threshold critical percentage mlr-percentage
```

For example,

```
[edit services video-monitoring templates t1]
user@host# set media-loss-rate threshold info percentage 5
user@host# set media-loss-rate threshold warning percentage 10
user@host# set media-loss-rate threshold critical percentage 20
```

7. Set the media rate variation thresholds for syslog message levels. The threshold is based on the ratio of the *difference* between the configured media rate and the monitored media rate to the configured media rate, expressed as a percentage.

```
[edit services video-monitoring templates template-name]
user@host# set media-rate-variation threshold info mrv-variation
user@host# set media-rate-variation threshold warning mrv-variation
user@host# set media-rate-variation threshold critical mrv-variation
```

For example,

```
[edit services video-monitoring templates t1]
user@host# set media-rate-variation threshold info 10
user@host# set media-rate-variation threshold warning 15
user@host# set media-rate-variation threshold critical 20
```

Configuring Interface Flow Criteria

You can identify the input and output flows that you want to monitor. If you do not specify any identifiers, all flows on the interface are monitored. Starting in Junos OS Release 17.2R1, you can identify IPv4-over-MPLS flows. Starting in Junos OS Release 17.4R1, you can identify IPv6 flows and IPv6-over MPLS flows. MPLS flows with more than three labels cannot be monitored.

> **NOTE:** You can configure a maximum of 256 flow definitions for an interface. If your flow definitions contain lists of addresses and ports, you can exceed the number of match conditions. When you exceed the limits for flows or match conditions, you receive the following constraint message when you commit:
>
> ```
> 'interfaces xe-0/2/2.0'
>    Number of flows or Number of match condition under flows exceeded limit
> error: configuration check-out failed
> ```

To configure monitoring of flows for interfaces:

1. In edit mode, identify an interface for monitoring.

   ```
   user@host# edit services video-monitoring interfaces interface-name
   ```

2. Identify IPv4 input flows for monitoring.

   a. Assign a name to the input flow.

      ```
      [edit services video-monitoring interfaces interface-name family inet]
      user@host# set input-flows input-flow-name
      ```

   b. Identify the source IP address or prefix value for the flow. You can use up to 32 addresses.

      ```
      [edit services video-monitoring interfaces interface-name family inet]
      user@host# set input-flows input-flow-name source-address [ address ]
      ```

   c. Identify the destination IP address or prefix value for the flow. You can use up to 32 addresses.

      ```
      [edit services video-monitoring interfaces interface-name family inet]
      user@host# set input-flows input-flow-name destination-address [ address ]
      ```

d. Identify the source port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set input-flows input-flow-name source-port [ port ]
```

e. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set input-flows input-flow-name destination-port [ port ]
```

f. Identify the template used to monitor the input flow on the interface.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set input-flows input-flow-name template template-name
```

3. Identify IPv4 output flows for monitoring.

a. Assign a name to the output flow.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set output-flows output-flow-name
```

b. Identify the source IP address or prefix value for the flow. You can use up to 32 addresses.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set output-flows output-flow-name source-address [ address ]
```

c. Identify the destination IP address or prefix value for the flow. You can use up to 32 addresses.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set output-flows output-flow-name destination-address [ address ]
```

d. Identify the source port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set output-flows output-flow-name source-port [ port ]
```

e. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set output-flows output-flow-name destination-port [ port ]
```

f. Identify the template used to monitor the output flow on the interface.

```
[edit services video-monitoring interfaces interface-name family inet]
user@host# set output-flows output-flow-name template template-name
```

4. Identify IPv4-over-MPLS input flows for monitoring:

   a. Assign a name to the input flow.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set input-flows input-flow-name
```

   b. Identify the payload type as IPv4 over MPLS.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set input-flows input-flow-name payload-type ipv4
```

   c. Identify the destination IP address or prefix value, the source IP address or prefix value, or both for the flow. You can use up to 32 destination addresses and up to 32 source addresses.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set input-flows input-flow-name destination-address [address]
user@host# set input-flows input-flow-name source-address [address]
```

   If you configure multiple addresses for both the destination and source, then either all the destination or all the source values must have the same prefix length. For example, the following is allowed, because all the destination addresses have the same prefix length.

```
[edit services video-monitoring interfaces ge-0/2/2.0 family mpls]
user@host# set input-flows input-flow-name destination-address [203.0.13.0/24 198.51.100.0/24]
user@host# set input-flows input-flow-name source-address [172.16.0.0/12 192.0.2.11/32]
```

   d. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set input-flows input-flow-name destination-port [ port ]
```

e. Identify the source port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set input-flows input-flow-name source-port [ port ]
```

f. Identify the template used to monitor the input flow on the interface.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set input-flows input-flow-name template template-name
```

5. Identify IPv4-over-MPLS output flows for monitoring:

a. Assign a name to the output flow.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set output-flows output-flow-name
```

b. Identify the payload type as IPv4 over MPLS.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set output-flows output-flow-name payload-type ipv4
```

c. Identify the destination IP address or prefix value, the source IP address or prefix value, or both for the flow. You can use up to 32 destination addresses and up to 32 source addresses.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set output-flows output-flow-name destination-address [address]
user@host# set output-flows output-flow-name source-address [address]
```

If you configure multiple addresses for both the destination and source, then either all the destination or all the source values must have the same prefix length. For example, the following is allowed, because all the destination addresses have the same prefix length.

```
[edit services video-monitoring interfaces ge-0/2/2.0 family mpls]
user@host# set output-flows output-flow-name destination-address [203.0.13.0/24 198.51.100.0/24]
user@host# set output-flows output-flow-name source-address [172.16.0.0/12 192.0.2.11/32]
```

d. Identify the source port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set output-flows output-flow-name source-port [port]
```

e. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set output-flows output-flow-name destination-port [ port ]
```

f. Identify the template used to monitor the output flow on the interface.

```
[edit services video-monitoring interfaces interface-name family mpls]
user@host# set output-flows output-flow-name template template-name
```

6. Identify IPv6 input flows for monitoring.

a. Assign a name to the input flow.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set input-flows input-flow-name
```

b. Identify the source IP address or prefix value for the flow. You can use up to 32 addresses.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set input-flows input-flow-name source-address [ address ]
```

c. Identify the destination IP address or prefix value for the flow. You can use up to 32 addresses.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set input-flows input-flow-name destination-address [ address ]
```

d. Identify the source port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set input-flows input-flow-name source-port [ port ]
```

e. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set input-flows input-flow-name destination-port [ port ]
```

f. Identify the template used to monitor the input flow on the interface.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set input-flows input-flow-name template template-name
```

7. Identify IPv6 output flows for monitoring.

a. Assign a name to the output flow.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set output-flows output-flow-name
```

b. Identify the source IP address or prefix value for the flow. You can use up to 32 addresses.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set output-flows output-flow-name source-address [ address ]
```

c. Identify the destination IP address or prefix value for the flow. You can use up to 32 addresses.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set output-flows output-flow-name destination-address [ address ]
```

d. Identify the source port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set output-flows output-flow-name source-port [ port ]
```

e. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set output-flows output-flow-name destination-port [ port ]
```

f. Identify the template used to monitor the output flow on the interface.

```
[edit services video-monitoring interfaces interface-name family inet6]
user@host# set output-flows output-flow-name template template-name
```

8. Identify IPv6-over-MPLS input flows for monitoring:

   a. Assign a name to the input flow.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set input-flows input-flow-name
   ```

   b. Identify the payload type as IPv6 over MPLS.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set input-flows input-flow-name payload-type ipv6
   ```

   c. Identify the destination IP address or prefix value, the source IP address or prefix value, or both for the flow. You can use multiple addresses (up to 32) for either the destination or the source IP address, but not for both.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set input-flows input-flow-name destination-address [address]
   user@host# set input-flows input-flow-name source-address [address]
   ```

   d. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set input-flows input-flow-name destination-port [ port ]
   ```

   e. Identify the source port for the flow. You can use multiple port numbers and port ranges.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set input-flows input-flow-name source-port [ port ]
   ```

   f. Identify the template used to monitor the input flow on the interface.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   ```

```
user@host# set input-flows input-flow-name template template-name
```

9. Identify IPv6-over-MPLS output flows for monitoring:

   a. Assign a name to the output flow.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set output-flows output-flow-name
   ```

   b. Identify the payload type as IPv6 over MPLS.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set output-flows output-flow-name payload-type ipv6
   ```

   c. Identify the destination IP address or prefix value, the source IP address or prefix value, or both for the flow. You can use multiple addresses (up to 32) for either the destination or the source IP address, but not for both.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set output-flows output-flow-name destination-address [address]
   user@host# set output-flows output-flow-name source-address [address]
   ```

   d. Identify the source port for the flow. You can use multiple port numbers and port ranges.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set output-flows output-flow-name source-port [port]
   ```

   e. Identify the destination port for the flow. You can use multiple port numbers and port ranges.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set output-flows output-flow-name destination-port [ port ]
   ```

   f. Identify the template used to monitor the output flow on the interface.

   ```
   [edit services video-monitoring interfaces interface-name family mpls]
   user@host# set output-flows output-flow-name template template-name
   ```

Configuring the Number of Flows That Can Be Measured

Starting in Junos OS Release 16.1R1, you can configure the number of flows that can be measured per Packet Forwarding Engine at a given time by an MPC. This value takes effect the next time the MPC is rebooted. If you do not configure this value, the default maximum value for an MPC is given in "Understanding Inline Video Monitoring on MX Series Routers" on page 827.

To configure the number of flows that can be measured per Packet Forwarding Engine by an MPC at a given time:

- Configure the flow table size. The range is 16 through 8192.

```
[edit chassis fpc slot inline-video-monitoring]
user@host# set flow-table-size size
```

Release History Table

Release	Description
17.4R1	Starting in Junos OS Release 17.4R1, you can identify IPv6 flows and IPv6-over MPLS flows.
17.2R1	Starting in Junos OS Release 17.2R1, you can identify IPv4-over-MPLS flows.
16.1R1	Starting in Junos OS Release 16.1R1, you can configure the number of flows that can be measured per Packet Forwarding Engine at a given time by an MPC.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

templates | 1172

interfaces | 1005

Inline Video Monitoring Syslog Messages on MX Series Routers

The following examples show the syslog messages produced when configured video monitoring thresholds are exceeded.

/var/log/messages

```
Mar 11 18:36:25  tstrtr01 fpc2 [MDI] DF: 56.71 ms, exceeded threshold for
flow(src:192.0.2.2 dst:198.51.100.2 sport:1024 dport:2048) ingressing at interface
 xe-2/2/1.0 with template t1.
Mar 11 18:36:25  tstrtr01 fpc2 [MDI] MLR : 112, exceeded threshold for flow
(src:192.0.2.2 dst:198.51.100.2 sport:1024 dport:2048) ingressing at interface
xe-2/2/1.0 with template t1.
Mar 11 18:36:25  tstrtr01 fpc2 [MDI] MRV : -5.67, exceeded threshold for flow
(src:192.0.2.2 dst:198.51.100.2 sport:1024 dport:2048) ingressing at interface
xe-2/2/1.0 with template t1.
```

Console Messages

```
NPC2(tstrtr01 vty)# [Mar 12 01:40:58.411 LOG: Critical] [MDI] MLR : 420, exceeded
 threshold for flow (src:192.0.2.2 dst:198.51.100.2 sport:1024 dport:2048)
ingressing at interface xe-2/2/1.0 with template t1.
[Mar 12 01:40:58.411 LOG: Critical] [MDI] MRV : -14.89, exceeded threshold for
flow (src:192.0.2.2 dst:198.51.100.2 sport:1024 dport:2048) ingressing at interface
 xe-2/2/1.0 with template t1.
[Mar 12 01:40:59.412 LOG: Critical] [MDI] DF: 141.74 ms, exceeded threshold for
flow(src:192.0.2.2 dst:198.51.100.2 sport:1024 dport:2048) ingressing at interface
 xe-2/2/1.0 with template t1.
```

RELATED DOCUMENTATION

Configuring Inline Video Monitoring on MX Series Routers | 833

Generation of SNMP Traps and Alarms for Inline Video Monitoring on MX Series Routers

Starting in Junos OS Release 15.1, SNMP support is introduced for the Media Delivery Index (MDI) metrics of inline video monitoring. Inline video monitoring is available on MX Series routers using only MPCE1, MPC2, MPC2E, MPC2E-NG, MPC5E, MPC6E, MPC7E, MPC8E, and MPC- 16XGE. Until Junos OS Release 14.2, inline MDI generated only syslogs when the computed MDI metric value was not within the configured range. SNMP support is now added to enable SNMP traps to be triggered when the computed delay factor (DF), media rate variation (MRV), and media loss rate (MLR) value is not within the configured range. You can retrieve the MDI statistics, flow levels, error details, and MDI record-level information using SNMP Get and Get Next requests. The SNMP traps and alarms that are generated when the MDI metrics exceed the configured ranges can be cleared as necessary. Also, you can control the flooding of SNMP traps on the system.

The following sections describe the statistical counters and parameters that are collected for MDI records and for generation of SNMP traps and alarms when the DF, MRV, and MLR values are not within the specified ranges.

Collection of MDI Statistics Associated with an FPC Slot

The FPC-level statistics include the following parameters that are displayed in the output of the **show services video-monitoring mdi stats fpc-slot *fpc-slot*** command. All of these attributes can be obtained using the SNMP Get request.

Table 109: show services video-monitoring mdi stats fpc-slot Output Fields

Field Name	Field Description
FPC Slot	Slot number of the monitored FPC
Active Flows	Number of active flows currently monitored. active flows = inserted flows - deleted flows.
Total Inserted Flows	Number of flows initiated under video monitoring.
Total Deleted Flows	Number of flows deleted due to inactivity timeout.
Total Packets Count	Number of total packets monitored.

Table 109: show services video-monitoring mdi stats fpc-slot Output Fields *(continued)*

Field Name	Field Description
Total Bytes Count	Number of total bytes monitored.
DF Alarm Count	Number of delay factor alarms at each of the following levels: • **Info level** • **Warning level** • **Critical level**
MLR Alarm Count	Number of media loss rate (MLR) alarms at each of the following levels: • **Info level** • **Warning level** • **Critical level**
MRV alarm count	Number of media rate variation (MRV) alarms at each of the following levels: • **Info level** • **Warning level** • **Critical level**

Collection of MDI Errors Associated with an FPC Slot

The FPC-level statistics include the following parameters that are displayed in the output of the **show services video-monitoring mdi errors fpc-slot *fpc-slot*** command. All of these attributes can be obtained using the SNMP Get request.

Table 110: show services video-monitoring mdi errors fpc-slot Output Fields

Field Name	Field Description
FPC slot	Slot number of the monitored FPC.
Flow Insert Error	Number of errors during new flow insert operations.

Table 110: show services video-monitoring mdi errors fpc-slot Output Fields *(continued)*

Field Name	Field Description
Flow Policer Drops	Number of packets dropped by flow policer process. **NOTE:** New flows usually arrive within a very short time interval (1.5 microseconds). These errors do not represent the loss of entire flows, because subsequent packets in the flow can establish the flow. All packets are monitored after a flow has been established. Packet forwarding occurs independently of the video monitoring, and packets are not dropped due to video monitoring errors.
Unsupported Media Packets Count	Number of packets dropped because they are not media packets or they are unsupported media packets.
PID Limit Exceeded	Number of packets unmonitored because the process identifier (PID) limit exceeded has been exceeded. **NOTE:** The current PID limit is 6.

Collection of MDI Flows Associated with an FPC Slot

The FPC-level statistics include the following parameters that are displayed in the output of the **show services video-monitoring mdi flows fpc-slot *fpc-slot*** command. All of these attributes can be obtained using the SNMP Get request.

Table 111: show services mdi flows Output Fields

Field Name	Field Description
SIP	Source IP address
DIP	Destination IP address
SP	Source port
DP	Destination port
Di	Direction (I=Input, O=Output)
Ty	Type of flow
Last DF:MLR	Delay factor and media loss rate value of last media delivery index record

Table 111: show services mdi flows Output Fields *(continued)*

Field Name	Field Description
Avg DF:MLR	Average value of delay factor and media loss rate
Last MRV	Media rate variation value of last media delivery index record
Avg MRV	Average value of media rate variation
IFL	Interface name on which flow is receiving
Template Name	Name of template associated with flow

Collection of MDI Record-Level Metrics

The computed DF, MLR, and MRV counters of all valid MDI records of a flow that you can view by using the output of the show services video-monitoring mdi flow fpc-slot fpc-slot detail command can be obtained by using the SNMP Get request.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

SNMP Traps for Inline Video Monitoring Statistics on MX Series Routers

SNMP is primarily used to monitor alarms raised by the inline video monitoring feature. The alarms sent to a network management system (NMS) either to troubleshoot the problem quickly or to proactively diagnose degradation in video quality. The following SNMP traps or alarms are implemented with the Cleared, Info, Warning, and Critical severity levels. The Cleared severity level is used to indicate a normal condition and to clear a particular alarm. Whenever a change in the alarm level occurs, the corresponding alarm is generated.

All the alarms include the following information pertaining to the MDI flows:

- Source IP address
- Destination IP address
- Source Port Destination

- Port Traffic type (UDP or RTP)
- Computed DF, MLR, and MRV values

The following traps are generated for MDI metrics:

- mdiMLRAlarm—This trap is generated when the computed MLR value is not within the configured range.
- mdiDFAlarm—This trap is generated when the computed DF value is not within the configured range.
- mdiMRVAlarm—This trap is generated when the computed MRV value is not within the configured range.

To enable the generation of SNMP traps or alarms for inline video monitoring or MDI metrics, include the **alarms** statement and its substatements at the **[edit services video-monitoring]** hierarchy level.

RELATED DOCUMENTATION

Processing SNMP GET Requests for MDI Metrics on MX Series Routers

A query on-demand mechanism without caching facility is used to process the SNMP Get requests. The Routing Engine queries the Packet Forwarding Engine to obtain the computed metrics on every Get request. The Routing Engine does not maintain computed metrics locally. No additional memory is required to cache queried metrics. The network management system (NMS) server can receive latest information on every Get request, especially regarding the MDI records because MDI records are updated very frequently. However, querying the Packet Forwarding Engine PFE on each GET request is resource-consuming if the volume of metrics is large. The response to a Get request might be relatively delayed as the Routing Engine has to poll the Packet Forwarding Engine to obtain the metrics.

Inline MDI metrics are real-time data where cached information might not be valid. Reporting cached or invalid metrics is not beneficial because it a real-time monitoring feature. An increase in the number of flows and number of MDI records per flow causes a proportional increase in the volume of memory required in the Routing Engine to store flows and MDI records for all flows. Because asynchronous traps are generated for threshold with enough contents, frequent Get request from NMS are not highly expected, reducing the periodicity of polling to the Packet Forwarding Engine. SNMP traps are triggered with the severity level of Info, Warning, Critical, or Cleared. A trap with the cleared severity level is used to clear an alarm.

Whenever a change in the alarm level occurs, the designated trap is triggered. For example, if the delay factor (DF) alarm changes from informational level to warning level, or from warning to critical, the mdiDFAlarm trap is triggered. Alarm can be immediate or average. If the immediate alarm is configured, an immediate trap is raised at the end of interval duration if the metric value exceeds the configured range.

If the average alarm is configured, a trap is generated, based on the average value for specified number of interval duration.

Storm control is applied for SNMP traps at the flow level and not at the FPC level. The NMS system can obtain SNMP trap from all the flows even if multiple flows are generating traps at approximately the same time. If multiple flows are generating traps at nearly the same time, NMS is flooded by many traps at the same time. For example, no traffic received on a logical interface owing to any reason can trigger all alarms and cause an avalance of alarms on the NMS server.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

PART 6

Configuration Statements and Operational Commands

CHAPTER 19

Configuration Statements

IN THIS CHAPTER

accounting

Syntax

```
accounting name {
  output {
    aggregate-export-interval seconds;
    cflowd hostname {
      aggregation {
        autonomous-system;
        destination-prefix;
        protocol-port;
        source-destination-prefix {
          caida-compliant;
        }
        source-prefix;
      }
      autonomous-system-type (origin | peer);
      port port-number;
      version format;
    }
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    interface interface-name {
      engine-id number;
      engine-type number;
      source-address address;
    }
  }
}
```

Hierarchy Level

```
[edit forwarding-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the discard accounting instance name and options.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Discard Accounting | 390

address (Interfaces)

Syntax

```
address address {
   destination address;
}
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-numberfamily family]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the interface address.

Options

address—Address of the interface.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Junos OS Network Interfaces Library for Routing Devices for other options not associated with flow monitoring.

Configuring Flow Monitoring | 3

Configuring Traffic Sampling on MX, M and T Series Routers | 375

address (Services Dynamic Flow Capture)

Syntax

```
address address;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name content-destination identifier]
```

Release Information
Statement introduced in Junos OS Release 7.4.

Description
Configure an IP address for the flow capture destination.

Options
address—IP address for the content destination.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Content Destination | 323

aggregate-export-interval

Syntax

```
aggregate-export-interval seconds;
```

Hierarchy Level

```
[edit forwarding-options accounting name output],
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls) output],
[edit forwarding-options sampling family (inet |inet6 |mpls) output]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the duration, in seconds, of the interval for exporting aggregate accounting information.

Options

seconds—Duration.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Discard Accounting | 390

aggregation

Syntax

```
aggregation {
    autonomous-system;
    destination-prefix;
    protocol-port;
    source-destination-prefix {
        caida-compliant;
    }
    source-prefix;
}
```

Hierarchy Level

[edit forwarding-options **accounting output cflowd** *hostname*],
[edit forwarding-options **sampling instance** *instance-name* **family** (inet |inet6 |mpls) **output flow-server** *hostname*],
[edit forwarding-options **sampling family** (inet |inet6 |mpls) **output flow-server** *hostname*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For cflowd version 8 only, specify the type of data to be aggregated; cflowd records and sends only those flows that match the specified criteria.

Options

autonomous-system—Aggregate by autonomous system (AS) number.

caida-compliant—Record source and destination mask-length values in compliance with the Version 2.1b1 release of CAIDA's cflowd application. If this statement is not configured, the Junos OS records source and destination mask length values in compliance with the *cflowd Configuration Guide*, dated August 30, 1999.

destination-prefix—Aggregate by destination prefix.

protocol-port—Aggregate by protocol and port number.

source-destination-prefix—Aggregate by source and destination prefix.

source-prefix—Aggregate by source prefix.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

alarms

Syntax

```
alarms {
    delay-factor {
        no-syslog-generation;
        generate-snmp-traps;
        storm-control {
            count number;
            interval number;
        }
        alarm-mode {
            mdi-records-count number;
            average;
        }
    }
    media-rate-variation {
        no-syslog-generation;
        generate-snmp-traps;
        storm-control {
            count number;
            interval number;
        }
        alarm-mode {
            mdi-records-count number;
            average;
        }
    }
    media-loss-rate {
        no-syslog-generation;
        generate-snmp-traps;
        storm-control {
            count number;
            interval number;
        }
        alarm-mode {
            immediate;
        }
    }
}
```

Hierarchy Level

[edit services]

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure the alarm to monitor and report active alarms. SNMP is used to monitor alarms raised by the inline video monitoring feature. The alarms are monitored in the network management system either to troubleshoot the problem or to diagnose degradation in video quality.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding Inline Video Monitoring on MX Series Routers | 827
- delay-factor | 911
- no-syslog-generation | 1059
- generate-snmp-traps | 983
- storm-control | 1153
- alarm-mode | 871
- media-rate-variation | 1037
- media-loss-rate | 1036

alarm-mode

Syntax

```
alarm-mode {
    mdi-records-count number;
    average;
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

If this statement is configured you can set the alarm as immediate or average mode. If immediate alarm is configured, an immediate trap is raised at the end of interval duration when the metric value exceeds the configured range. If average alarm is configured, a trap is generated based on average value for the specified number of interval duration.

Default

The default alarm mode is immediate mode.

Options

mdi-records-count *number*—Use the specified media delivery index record count number for immediate alarm mode.

average—Generate traps for average values that are not within the configured range.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

alarms | 869

allowed-destinations

Syntax

```
allowed-destinations [ destinations ];
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name control-source identifier]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Identify flow capture destinations that are allowed in messages sent from this control source.

Options

destinations—Allowed content destination name.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

analyzer-address

Syntax

```
analyzer-address address;
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure an IP address for the packet analyzer that overrides the default value.

Options

address—IP address for packet analyzer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring a Packet Analyzer | 272

analyzer-id

Syntax

```
analyzer-id name;
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure an identifier for the packet analyzer that overrides the default value.

Options
name—Identifier for packet analyzer.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring a Packet Analyzer | 272

archive-sites

Syntax

```
archive-sites {
    ftp:url {
        password "password";
        username username;
    }
}
```

Hierarchy Level

[edit services flow-collector **transfer-log-archive**]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the destination for transfer logs.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

authentication-mode

Syntax

```
authentication-mode (authenticated | encrypted | none);
```

Hierarchy Level

```
[edit services rpm twamp server],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced in Junos OS Release 9.5.

Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the authentication or encryption mode support for the TWAMP test protocol. This statement is required in the configuration; if no authentication or encryption is specified, you must set the value to **none**.

Options

authenticated—Authenticate all TWAMP packets.

> **NOTE:** This mode is supported only on TWAMP servers.

encrypted—Encrypt all TWAMP packets.

> **NOTE:** This mode is supported only on TWAMP servers.

none—Do not authenticate or encyrpt packets.

> **NOTE:** This mode is supported on both TWAMP servers and clients.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610
Understanding Two-Way Active Measurement Protocol on Routers | 582

authentication-key-chain (TWAMP)

Syntax

```
authentication-key-chain identifier {
    key-id identifier {
        secret password-string;
    }
}
```

Hierarchy Level

[edit services rpm twamp server]

Release Information

Statement introduced in Junos OS Release 9.5.

Description

Apply and enable an authentication keychain to the routing device. Note that the referenced key chain must be defined. When configuring the authentication key update mechanism for TWAMP, you cannot commit the **0.0.0.0/allow** statement with authentication keys or key chains. The CLI issues a warning and fails to commit such configurations.

Options

identifier—Authentication keychain name. It can be up to 126 characters. Characters can include any ASCII strings. If you include spaces, enclose all characters in quotation marks (“ ”).

password-string—Authentication key, consisting of 1 through 8 ASCII characters. If the key contains spaces, enclose it in quotation marks.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

autonomous-system-type

Syntax

```
autonomous-system-type (origin | peer);
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls) output flow-server hostname],
[edit forwarding-options sampling family (inet |inet6 |mpls) output flow-server hostname]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.

Description

Specify the type of AS numbers that cflowd exports**.**

Default

origin

Options

origin—Export origin AS numbers of the packet source address in the Source Autonomous System cflowd field.

peer—Export peer AS numbers through which the packet passed in the Source Autonomous System cflowd field.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

bandwidth-kbps (RFC 2544 Benchmarking)

Syntax

```
bandwidth-kbps kbps;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarking profilestest-profile profile-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.

Description

Define the theoretical maximum bandwidth, in kilobits per second, for the test. The theoretical limit of the media for the frame size configured for the test. This value is typically set to the bandwidth of the server being tested. The range is 1,000 Kbps through 1,000,000 Kbps (1 Gbps). The value defined is the highest bandwidth value used for this test.

Options

kbps—Bandwidth limit, in kilobits per second (kpbs).

Range: 1,000 kbps through 1,000,000 Kbps.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- RFC 2544-Based Benchmarking Tests Overview | **763**
- Configuring RFC 2544-Based Benchmarking Tests | **770**
- rfc2544-benchmarking | **1108**

bgp

Syntax

```
bgp {
    data-fill data;
    data-size size;
    destination-port port;
    history-size size;
    logical-system logical-system-name <routing-instances routing-instance-name>;
    moving-average-size size;
    probe-count count;
    probe-interval seconds;
    probe-type type;
    routing-instances instance-name;
    rfc6514-compliant-safi129;
    test-interval interval;
}
```

Hierarchy Level

```
[edit services rpm bgp],
[edit protocols bgp group group-name],
[edit routing-instances instance-name protocols bgp group group-name],
[edit logical-system logical-system-name protocols bgp group group-name],
[edit logical-system logical-system-name routing-instances instance-name protocols bgp group group-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure BGP neighbor discovery through Real-Time Performance Monitoring (RPM).

Options

bgp—Define properties for configuring BGP neighbor discovery.

The remaining statements are explained separately. See CLI Explorer.

> **NOTE:** On MX Series routers, you can configure all the statements. On M Series and T Series routers, you can configure only the **logical-system** and **routing-instances** statements.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

bridge-template

Syntax

```
bridge-template;
```

Hierarchy Level

```
[edit services flow-monitoring version-ipfix |version9 template template-name]
```

Release Information
Statement introduced in Junos OS Release 18 .2R1.

Description
Specify that the template is used for bridge records or for VPLS records.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

capture-group

Syntax

```
capture-group client-name {
    content-destination identifier {
        address address;
        hard-limit bandwidth;
        hard-limit-target bandwidth;
        soft-limit bandwidth;
        soft-limit-clear bandwidth;
        ttl hops;
    }
    control-source identifier {
        allowed-destinations [ destinations ];
        minimum-priority value;
        no-syslog;
        notification-targets address port port-number;
        service-port port-number;
        shared-key value;
        source-addresses [ addresses ];
    }
    duplicates-dropped-periodicity seconds;
    input-packet-rate-threshold rate;
    interfaces interface-name;
    max-duplicates number;
    pic-memory-threshold percentage percentage;
}
```

Hierarchy Level

```
[edit services dynamic-flow-capture]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Define the capture group values.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Capture Group | 322

cflowd (Discard Accounting)

Syntax

```
cflowd hostname {
    aggregation {
        autonomous-system;
        destination-prefix;
        protocol-port;
        source-destination-prefix {
            caida-compliant;
        }
        source-prefix;
    }
    autonomous-system-type (origin | peer);
    label-position {
        template template-name;
    }
    (local-dump | no-local-dump);
    port port-number;
    source-address address;
    version format;
}
```

Hierarchy Level

```
[edit forwarding-options accounting name output]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Collect an aggregate of sampled flows and send the aggregate to a specified host system that runs the collection utility cfdcollect.

You can configure up to one version 5 and one version 8 flow format at the **[edit forwarding-options accounting *name* output]** hierarchy level.

Options

hostname—IP address or identifier of the host system (the workstation running the cflowd utility).

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

cflowd (Flow Monitoring)

Syntax

```
cflowd hostname {
    port port-number;
}
```

Hierarchy Level

```
[edit forwarding-options monitoring name inet output]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Collect an aggregate of sampled flows and send the aggregate to a specified host system that runs the collection utility cfdcollect.

You can configure up to eight version 5 flow formats at the **[edit forwarding-options monitoring *name* output]** hierarchy level. Version 8 flow formats are not supported for flow-monitoring applications.

Options

hostname—IP address or identifier of the host system (the workstation running the cflowd utility).

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

client

Syntax

```
client {
   control-connection control-client- name {
      authentication-mode
      destination-interface interface-name;
      destination-port port;
      history-size size;
      moving-average-size number;
      persistent-results
      routing-instance instance-name;
      target (url url | address address);
      tcp-keepcnt
      tcp-keepidle
      tcp-keepintvl
      test-interval interval;
      traps traps;
      data-fill-with zeros
      data-size size;
      dscp-code-point dscp-bits;
      probe-count count;
      probe-interval seconds;
      thresholds thresholds;
      test-session session-name{
         data-fill-with zeros data;
         data-size size;
         dscp-code-point dscp-bits;
         probe-count count;
         probe-interval seconds;
         target (url url | address address);
      }
   }
}
```

Hierarchy Level

```
[edit services rpm twamp]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the TWAMP client configuration settings.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Two-Way Active Measurement Protocol on Routers | 582

client-delegate-probes

Syntax

```
rpm client-delegate-probes;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number]
```

Release Information

Statement introduced in Junos OS Release 17.3R1 on MX Series routers.

Description

Generate real-time performance monitoring (RPM) probes on an MS-MPC or MS-MIC services interface, which increases the number of RPM probes that can run at the same time.

The **destination-interface** statement must be configured at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level to point to the interface and logical unit number and for which you configure **client-delegate-probes**. Configure the **delegate-probes** statement at the **[edit services rpm probe *owner*]** hierarchy level to complete the configuration.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

client-list

Syntax

```
client-list list-name {
    address address;
}
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Specify the list of allowed control client hosts that can connect to this server. Each entry is a Classless Interdomain Routing (CIDR) address (IP address plus mask) that represents a network of allowed hosts. You can configure more than one list, but you must configure at least one client address to enable TWAMP. Each list can contain up to 64 entries.

Options

list-name—Name of client address list.

address—Address and mask for an allowed client.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

collector

Syntax

```
collector interface-name;
```

Hierarchy Level

```
[edit services flow-collector interface-map]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the default flow collector interface for interface mapping.

Options

interface-name—Default flow collector interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Interface Mappings | 273

collector (Inline Monitoring)

Syntax

```
collector name {
    destination-address destination-ip-address;
    destination-port destination-port;
    dscp dscp;
    forwarding-class forwarding-class;
    routing-instance routing-instance;
    sampling-rate sampling-rate;
    source-address source-ip-address;
}
```

Hierarchy Level

```
[edit services inline-monitoring instance]
```

Release Information

Statement introduced in Junos OS 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Configure an collector for inline monitoring. The monitored packets are exported to the collector in an IPFIX format. The actual packet is exported in an IPFIX format up to the configured clip length. By default, Junos OS supports a maximum packet length of 126 bytes beginningstarting with the Ethernet header. The IPFIX format exports information on the original packet size, and incoming or outgoing interface for further processing on the collector.

Options

name—Name of collector.

destination-address—IPv4 destination IP address.

destination-port—Destination port value.

Range: 1 through 65535

dscp—DSCP value.

Default: 0

Range: 0 through 63

forwarding-class—Forwarding class for exported frames.

Default: best-effort

routing-instance—Name of routing instance.

Default: default.inet

sampling-rate—Sampling rate.

Range: 1 through 16000000

Default: 1

source-address—IPv4 source IP address.

Required Privilege Level
system

RELATED DOCUMENTATION

Understanding Inline Monitoring Services | 362

collector (Flow Monitoring Logs for NAT)

Syntax

```
collector collector-name {
    source-ip address;
    destination-address address;
    destination-port port-number;
}
```

Hierarchy Level

```
[edit services jflow-log]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the name of the collector to which flow monitoring log messages in IPFIX or version 9 flow template format for NAT events must be sent. The generated flow monitoring logs for NAT events in flow template format are sent to the specified host or external device that functions as the NetFlow collector. You must associate a collector with a template profile for the template characteristics, such as refresh rate of messages and the template format, to be used for generated flow monitoring logs.

Options

collector-name—Name of the collector to which flow monitoring log messages for NAT events in flow monitoring format (IPFIX or version 9 flow template format) must be sent. The name can be up to 32 alphanumeric characters in length. Allowed characters are [a-zA-Z0-9_]

The remaining statements are described separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**

Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**

collector (Flow Template Profiles for NAT)

Syntax

```
collector collector-name;
```

Hierarchy Level

```
[edit services jflow-log template-profile template–profile-name]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the name of the collector to be associated with a template profile. The generated flow monitoring logs for NAT events in flow template format are sent to the specified collector. You must have previously configured the collector by using the **collector collector-name** statement at the **[edit services jflow-log]** hierarchy level before you associate a collector with a template profile.

Options

collector-name—Name of the collector to which flow monitoring log messages for NAT events in flow monitoring format (IPFIX or version 9 flow template format) must be sent. The name can be up to 32 alphanumeric characters in length. Allowed characters are [a-zA-Z0-9_]

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

collector-group (Flow Template Profiles for NAT)

Syntax

collector-group *collector-group-name*;

Hierarchy Level

[edit services jflow-log template-profile *template–profile-name*]

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the name of the collector group to be associated with a template profile. The generated flow monitoring logs for NAT events in flow template format are sent to the specified collector group. By using a collector group, you can effectively and optimally transmit flow monitoring logs to a cluster of collectors in a single, one-step operation. A maximum of up to eight collectors can be aggregated into a collector group. You must have previously configured the collector group by using the **collector-group collector-group-name** statement at the **[edit services jflow-log]** hierarchy level before you associate a collector-group with a template profile.

Options

collector-group-name—Name of the collector group to which log messages for NAT events in flow monitoring format (IPFIX or version 9 flow template format) must be sent. The name can be up to 32 alphanumeric characters in length. Allowed characters are [a-zA-Z0-9_]

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

collector-group (Flow Monitoring Logs for NAT)

Syntax

```
collector-group collector-group-name {
    [collector-name1 collector-name2];
}
```

Hierarchy Level

```
[edit services jflow-log]
```

Release Information
Statement introduced in Junos OS Release 15.1.

Description

Specify the name of the collector group that contains a set of NetFlow collectors to which flow monitoring log messages in IPFIX or version 9 flow template format for NAT events must be sent. You must define at least one collector in the group. A maximum of up to eight collectors can be aggregated into a collector group.

The generated flow monitoring logs for NAT events in flow template format are sent to the specified collector group. By using a collector group, you can effectively and optimally transmit flow monitoring logs to a cluster of collectors in a single, one-step operation.

Options

collector-group-name—Name of the collector group to which flow monitoring log messages for NAT events in flow monitoring format (IPFIX or version 9 flow template format) must be sent. The name can be up to 32 alphanumeric characters in length. Allowed characters are [a-zA-Z0-9_]

collector-name—Name of the collector to be assigned to the group of collectors. You must have previously defined the collector by including the **collector *collector-name*** statement at the **[edit services jflow-log]** hierarchy level. You can specify a list of valid collector names. Specify the names individually by using a space to separate each collector name. The name can be up to 32 alphanumeric characters in length. Allowed characters are [a-zA-Z0-9_]

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

content-destination

Syntax

```
content-destination identifier {
    address address;
    hard-limit bandwidth;
    hard-limit-target bandwidth;
    soft-limit bandwidth;
    soft-limit-clear bandwidth;
    ttl hops;
}
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Identify the destination for captured packets.

Options

identifier—Name of the destination.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Content Destination | 323

control-connection

Syntax

```
control-connection control-client-name {
    authentication-mode
    destination-interface interface-name;
    destination-port port;
    history-size size;
    moving-average-size number;
    persistent-results
    routing-instance instance-name;
    target (url url | address address);
    tcp-keepcnt
    tcp-keepidle
    tcp-keepintvl
    test-interval interval;
    traps traps;
    data-fill-with zeros data;
    data-size size;
    dscp-code-point dscp-bits;
    probe-count count;
    probe-interval seconds;
    thresholds thresholds;
    test-session session-name{
        data-fill-with zeros data;
        data-size size;
        dscp-code-point dscp-bits;
        probe-count count;
        probe-interval seconds;
        target (url url | address address);
    }
}
```

Hierarchy Level

```
[edit services rpm twamp client]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

List all the TWAMP control clients that can connect to this server. You must configure at least one client to enable TWAMP.

Options

control-client-name—Name of the control client.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Two-Way Active Measurement Protocol on Routers | 582

control-source

Syntax

```
control-source identifier {
    allowed-destinations [ destinations ];
    minimum-priority value;
    no-syslog;
    notification-targets address port port-number;
    service-port port-number;
    shared-key value;
    source-addresses [ addresses ];
}
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Identify the source of the dynamic flow capture request.

Options

identifier—Name of control source.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

core-dump

Syntax

```
(core-dump | no-core-dump);
```

Hierarchy Level

```
[edit interfaces mo-fpc/pic/port multiservice-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

A useful tool for isolating the cause of a problem. Core dumping is enabled by default. The directory **/var/tmp** contains core files. Junos OS saves the current core file (0) and the four previous core files, which are numbered from 1 through 4 (from newest to oldest):

> **NOTE:** By default, all members of a configured user group (with read-only permissions) can access the core dump files and attach them to cases associated with JTAC.

- **core-dump**—Enable the core dumping operation.
- **no-core-dump**—Disable the core dumping operation.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

data-fill

Syntax

```
data-fill data;
data-fill-with-zeros data;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 9.3 for PTX Series Packet Transport routers.
Statement at the [edit services rpm twamp client control-connection *control-client-name*] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the contents of the data portion of Internet Control Message Protocol (ICMP) probes. The **data-fill** statement is not valid with the **http-get** or **http-metadata-get** probe types. For TWAMP client, if this knob is set, then fill the test packet with zeros, if the knob is not set then the data content is random value as indicated in RFC.

Options

data—A hexadecimal value; for example, **0-9**, **A-F**.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | **629**

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

data-fill-with zeros

Syntax

```
data-fill-with-zeros;
```

Hierarchy Level

[edit services rpm twamp client control-connection *control-client-name* test-session *session-name*]

Release Information

Statement introduced in Junos OS Release 15.1 for MX Series routers.

Description

If this statement is configured, then the contents of the test packet are zeros, if the statement is not configured, then the data content is a pseudo-random number.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Two-Way Active Measurement Protocol on Routers | 582

data-format

Syntax

```
data-format format;
```

Hierarchy Level

```
[edit services flow-collector file-specification variant variant-number]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the data format for a specific file format variant.

Options

format—Data format. Specify **flow-compressed** as the data format.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring File Formats | 272

data-size

Syntax

```
data-size size;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name test-session session-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the size of the data portion of ICMP probes. The **data-size** statement is not valid with the **http-get** or **http-metadata-get** probe type.

Options

size—0 through 65400 for RPM, for TWAMP the value is from 60 through 1400.

Default: 0 for RPM and 60 for TWAMP.

> **NOTE:** If you configure the hardware timestamp feature (see "Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches" on page 602):
>
> - The default value of **data-size** is 32 bytes and **32** is the minimum value for explicit configuration. The UDP timestamp probe type is an exception; it requires a minimum data size of 52 bytes.
> - The data size must be at least 100 bytes smaller than the default MTU of the interface of the RPM client interface.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring BGP Neighbor Discovery Through RPM | **629**

delay-factor

Syntax

```
delay-factor {
    no-syslog-generation;
    generate-snmp-traps;
    storm-control {
        count number;
        interval number;
    }
    alarm-mode {
        mdi-records-count number;
        average;
    }
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure the maximum observed time difference between the arrival of media data and the drain of media data. The delay factor suggests the minimum size of the buffer required at the next downstream node. As a stream progresses, the variation of the delay factor indicates packet bunching or packet gaps (jitter). Greater delay factor values also indicate that more network latency is needed to deliver a stream because of the need to pre-fill a receive buffer before beginning the drain to guarantee no underflow.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

alarms | 869

delegate-probes

Syntax

```
delegate-probes;
```

Hierarchy Level

[edit services rpm probe *owner*]

Release Information
Statement introduced in Junos OS Release 17.3R1 on MX Series routers.

Description
Generate real-time performance monitoring (RPM) probes on an MS-MPC or MS-MIC card, which increases the number of RPM probes that can run at the same time.

To use the **delegate-probes** statement, you must first configure the **destination-interface** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level to point to a valid logical unit number of a multiservices interface. Then configure the same unit and multiservice interface with the **rpm client-delegate-probes** statement at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

The **probe-type *type*** at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level can be **icmp-ping** or **icmp-ping-timestamp** starting in Junos OS Release 17.3R1, and **icmp6-ping** starting in Junos OS Release 18.1R1.

To avoid packet bursts in the network due to RPM, probes will be distributed in a better way.

The chances of multiple tests starting and ending at the same time are smaller. This way RPM syslog bursts and a potential performance bottleneck in **event-processing** are avoided.. This does not exclude potential syslog drops on the RE if more than 12000 RPM tests are running simultaneously. For scaled configurations (with more than 12000 RPM tests) we recommend you to configure syslogs to sent to an external hosts for offloaded processing.

> **NOTE:** You cannot configure the **routing-instance** statement at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level for RMP probes that are generated on an MS-MPC or MS-MIC card.

Required Privilege Level
interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

client-delegate-probes | **890**

destination (Interfaces)

Syntax

```
destination address;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number tunnel],
[edit interfaces interface-name unit logical-unit-number family inet address address],
[edit interfaces interface-name unit logical-unit-number tunnel],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number family inet address
  address]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For CoS on ATM interfaces, specify the remote address of the connection.

For point-to-point interfaces only, specify the address of the interface at the remote end of the connection.

For tunnel and encryption interfaces, specify the remote address of the tunnel.

Options

address—Address of the remote side of the connection.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- *Configuring Linear RED Profiles on ATM Interfaces*
- *Multilink and Link Services Logical Interface Configuration Overview*
- *Configuring Encryption Interfaces*
- Configuring Traffic Sampling on MX, M and T Series Routers | 375
- Configuring Flow Monitoring | 3
- *Configuring Unicast Tunnels*

destination-address (Flow Monitoring Logs for NAT)

Syntax

```
destination-address address;
```

Hierarchy Level

```
[edit services jflow-log collector collector-name]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the destination IP address or identifier of the host or external device that functions as the collector for receiving the generated flow monitoring logs that are sent from the exporter. You can configure an IPv4 address, or an identifier of the host system (the workstation either running the Jflow utility or collecting traffic flows using version 9 or IPFIX format). For external NetFlow collectors or servers, the hostname must be reachable from the same routing instance to which the initial data packet (that triggered session establishment) is delivered. You can specify a maximum of eight collectors per profile.

Options

address—Destination hostname, or IPv4 or IPv6 address of the collector.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

destination-interface

Syntax

```
destination-interface interface-name;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name],
[edit services rpm probe-server (tcp | udp)],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced in Junos OS Release 7.5.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

On M Series and T Series routers, specify a services (**sp-**) interface that adds a timestamp to RPM probe messages. This feature is supported only with **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, and **udp-ping-timestamp** probe types. You must also configure the **rpm** statement on the **sp-** interface and include the **unit 0 family inet** statement with a **/32** address.

On M Series, MX Series, and T Series routers, specify a multiservices (**ms-**) interface that adds a timestamp to RPM probe messages. This feature is supported only with **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, and **udp-ping-timestamp** probe types. You must also configure the **rpm** statement on the **ms-** interface and include the **unit 0 family inet** statement with a **/32** address.

The inline service interface (si- interface) is a virtual physical service interface that resides on the Packet Forwarding Engine to provide L2TP services without a special services PIC. The inline service interface is supported only by MPCs on MX Series routers. Four inline service interfaces are configurable per MPC-occupied chassis slot. Specify a multiservices (**si-**) interface that adds a timestamp to TWAMP probe messages. You must also configure the **rpm twamp-client** or **twamp-server** statement on the **si-** interface and include the **unit 0 family inet** statement with a **/32** address.

To enable RPM for the extension-provider packages on the adaptive services interface, configure the **object-cache-size**, **policy-db-size**, and **package** statements at the **[edit chassis fpc *slot-number* pic *pic-number* adaptive-services service-package extension-provider]** hierarchy level. For the extension-provider package, ***package-name*** in the **package *package-name*** statement is **jservices-rpm**.

Starting in Junos OS Release 17.3R1, you can use **destination-interface *interface-name.logical-unit-number*** at the **[edit services rpm probe *owner* test *test-name*]** hierarchy level to configure the generation of probes on an MS-MPC or MS-MIC. You must also inlcude the **delegate-probes** statement at the **[edit services**

rpm probe *owner*] hierarchy level and the **rpm client-delegate-probes** and the **family (inet | inet6) address *address*** statements at the **[edit interfaces *interface-name* unit *logical-unit-number*]** hierarchy level.

Options

interface-name—Name of the adaptive services interface.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches | **602**
- Configuring RPM Receiver Servers | **601**
- Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches | **602**
- hardware-timestamp | **986**
- rpm (Interfaces) | **1115**
- Enabling RPM on MX, M and T Series Routers and SRX Firewalls for the Services SDK | **641**

destination-ipv4-address (RFC 2544 Benchmarking)

Syntax

```
destination-ipv4-address address;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the destination IPv4 address to be used in generated test frames. You must configure this option if you specify inet as the family. This option is not required if you specify **ccc**as the family.

Options

address—Valid IPv4 address.

Default: If you do not configure the destination IPv4 address, the default value of 192.168.1.20 is used.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

destination-mac-address (RFC2544 Benchmarking)

Syntax

```
destination-mac-address mac-address;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Specify the destination MAC address used in the generated test frames. This is a mandatory parameter for family **bridge**.

Options

mac-address—MAC address. Specify the MAC address as six hexadecimal bytes in one of the following formats: ***nnnn.nnnn.nnnn*** or ***nn:nn:nn:nn:nn:nn***—for example, **0000:5e00:5355** or **00:00:5e:00:53:55**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- rfc2544-benchmarking | **1108**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- Configuring an RFC 2544-Based Benchmarking Test | **659**

destination-port

Syntax

```
destination-port port;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport Routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the User Datagram Protocol (UDP) or Transmission Control Protocol (TCP) port to which a probe is sent. This statement is used only for TCP or UDP probe types.

The value for the **destination-port** can be only 7 when you configure the destination port along with hardware timestamping. A constraint check prevents you for configuring any other value for the destination port in this case.

This constraint does not apply when you are using one-way hardware timestamping along with **destination-port** and either **probe-type udp-ping** or **probe-type udp-ping-timestamp**.

Options

Default: The default value for the port is 862 to which the TWAMP client establishes control connection.

port—Port number **7** or from **49,160** through **65,535**.

> **NOTE:** The specified port numbers are recommended for RPM only.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

destination-port (Flow Monitoring Logs for NAT)

Syntax

destination-port *port-number*;

Hierarchy Level

[edit services jflow-log collector *collector-name*]

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the UDP port of the destination to be used in the UDP header for the generated flow monitoring logs. This is a required setting.

Options

port-number—UDP port number for the test frames.

Default: 4041

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | 285

Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | 295

Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | 307

Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | 309

destination-udp-port (RFC 2544 Benchmarking)

Syntax

```
destination-udp-port port-number;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the UDP port of the destination to be used in the UDP header for the generated frames. If you do not specify the UDP port, the default value of 4041 is used.

Options

port-number—UDP port number for the test frames

Default: 4041

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

destinations

Syntax

```
destinations {
    ftp:url {
        password "password";
    }
}
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the primary and secondary destination FTP servers.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Destination FTP Servers for Flow Records | 271

direction (RFC2544 Benchmarking)

Syntax

```
direction (egress | ingress);
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the direction of the interface on which the test must be run. This parameter is valid only for a **ccc** family and a **bridge** family. RFC2544 tests are supported only in the egress direction or the user-to-network interface (UNI) direction of an E-line or E-LAN service parameters in a bridge domain between two routers for unicast traffic. You cannot compute the NNI direction of Ethernet services between two routers for multicast or broadcast traffic.

Options

egress—Run the test in the egress direction of the interface (network-to-network interface (NNI)). This option is applicable for a **ccc** and **bridge** family.

ingress—Run the test in the ingress direction of the interface (user-to-network interface (UNI)). You cannot configure this option for a **bridge** family.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

rfc2544-benchmarking | **1108**

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

Configuring an RFC 2544-Based Benchmarking Test | **659**

disable (Forwarding Options)

Syntax

```
disable;
```

Hierarchy Level

```
[edit forwarding-options port-mirror],
[edit forwarding-options port-mirror instance instance-name],
[edit forwarding-options sampling],
[edit forwarding-options sampling instance instance-name],
[edit forwarding-options sampling family (inet | inet6 | mpls | vpls) ],
[edit forwarding-options sampling family (inet | inet6 | mpls | vpls) output file]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement added to **port-mirror** hierarchy in Junos OS Release 9.6.

> **NOTE:** Beginning in Junos OS Release 15.1F5 and later 15.1 releases and Junos OS Release 16.1 and later, the **disable** option has been deprecated at the **forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls)** hierarchy level on PTX3000 Series routers. When configured, the option does not take effect, so packets continue to be sampled. Instead of the **disable** option, use the **deactivate forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls)** command to prevent sampling.

Description

Disable traffic accounting, port mirroring, or sampling.

> **NOTE:** The **disable** statement at the **[edit forwarding-options sampling]** hierarchy level disables only Routing Engine-based sampling. To disable PIC-based sampling and inline sampling, include the **disable** statement at the **[edit forwarding-options sampling instance *instance-name*]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Disabling Traffic Sampling

Configuring Traffic Sampling on MX, M and T Series Routers | 375

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

disable-signature-check (RFC 2544 Benchmarking)

Syntax

```
disable-signature-check;
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 15.1 for MX104 Universal Routing Platforms.

Description

Disable signature verification on the received test frames. This statement is valid only if you configure the test mode to be a reflector. The configuration is useful when the test traffic is generated using a third-party vendor tool, instead of an ACX Series router.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647

Supported RFC2544-Based Benchmarking Statements on MX Series Routers | 657

Configuring an RFC 2544-Based Benchmarking Test | 659

dscp (flow-server)

Syntax

```
dscp dscp-value
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet | inet6) output flow-server hostname]
```

Release Information

Statement introduced in Junos OS Release 15.1F4 for the PTX Series.
Statement introduced in Junos OS Release 16.1 for the MX Series.

Description

Specify the Differentiated Services Code Point (DSCP) mapping that is applied to exported packets for inline active flow monitoring. This allows different levels of service to be assigned to sampled traffic.

Options

dscp *dscp-value*—Can be a value between 0 and 63 (the default is 0). When the same **flow-server** is configured under both the **inet** and **inet6** families in a sampling instance, use the same **dscp** value for both **flow-server** appearances.

The *dscp-value* is overwritten by the CoS DSCP value if you configure **dscp** under the **[edit class-of-service]** hierarchy.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Inline Active Flow Monitoring on PTX Series Routers | 458
- Configuring Flow Aggregation to Use IPFIX Flow Templates on PTX Series Routers | 520
- Configuring Sampling Instance on MX, M and T Series Routers or QFX Series Switches | 388

dscp-code-point

Syntax

```
dscp-code-point dscp-bits;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name test-session session-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the value of the Differentiated Services (DiffServ) field within the IP header. The DiffServ code point (DSCP) bits value must be set to a valid 6-bit pattern.

Options

dscp-bits—A valid 6-bit pattern; for example, **001111**, or one of the following configured DSCP aliases:

- **af11**—Default: 001010
- **af12**—Default: 001100
- **af13**—Default: 001110
- **af21**—Default: 010010
- **af22**—Default: 010100
- **af23** —Default: 010110
- **af31** —Default: 011010
- **af32** —Default: 011100
- **af33** —Default: 011110
- **af41** —Default: 100010
- **af42** —Default:100100
- **af43** —Default:100110
- **be**—Default: 000000

- **cs1**—Default: 001000
- **cs2**—Default: 010000
- **cs3**—Default: 011000
- **cs4**—Default: 100000
- **cs5**—Default: 101000
- **cs6**—Default: 110000
- **cs7**—Default: 111000
- **ef**—Default: 101110
- **nc1**—Default: 110000
- **nc2**—Default: 111000

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**
Understanding Two-Way Active Measurement Protocol on Routers | **582**

dump-on-flow-control

Syntax

```
dump-on-flow-control;
```

Hierarchy Level

```
[edit interfaces interface-name multiservice-options]
```

Release Information
Statement introduced in Junos OS Release 9.5.

Description
This option supports high availability functionality and can be used with various service interfaces, including **rsp**, **rms**, **lsq**, and **rlsq**.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- *Configuring Multiservice Physical Interface Properties*
- *Junos OS Services Interfaces Library for Routing Devices*
- *passive-monitor-mode*

duplicates-dropped-periodicity

Syntax

```
duplicates-dropped-periodicity seconds;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify the frequency for sending notifications to affected control sources when transmission of duplicate sets of data is restricted because the **max-duplicates** threshold has been reached.

Options

seconds—Period for sending DuplicatesDropped notifications.

Default: 30 seconds

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- g-duplicates-dropped-periodicity | **981**
- Limiting the Number of Duplicates of a Packet | **328**
- max-duplicates | **1027**

dynamic-flow-capture

Syntax

```
dynamic-flow-capture {
    capture-group client-name {
        content-destination identifier {
            address address;
            hard-limit bandwidth;
            hard-limit-target bandwidth;
            soft-limit bandwidth;
            soft-limit-clear bandwidth;
            ttl hops;
        }
        control-source identifier {
            allowed-destinations [ destinations ];
            minimum-priority value;
            no-syslog;
            notification-targets address port port-number;
            service-port port-number;
            shared-key value;
            source-addresses [ addresses ];
        }
        duplicates-dropped-periodicity seconds;
        input-packet-rate-threshold rate;
        interfaces interface-name;
        max-duplicates number;
        pic-memory-threshold percentage percentage;
    }
    g-duplicates-dropped-periodicity seconds;
    g-max-duplicates number;
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Define the dynamic flow capture properties to be applied to traffic.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Junos Capture Vision | 319

engine-id (Forwarding Options)

Syntax

```
engine-id number;
```

Hierarchy Level

```
[edit forwarding-options accounting name output interface interface-name],
[edit forwarding-options monitoring name output interface interface-name],
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output interface interface-name],
[edit forwarding-options sampling family (inet | inet6 | mpls) output interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the engine ID number for flow monitoring and accounting services.

Options

number—Identity of accounting interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Traffic Sampling on MX, M and T Series Routers | 375
- Configuring Flow Monitoring | 3
- Configuring Discard Accounting | 390

engine-type

Syntax

```
engine-type number;
```

Hierarchy Level

```
[edit forwarding-options accounting name output interface interface-name],
[edit forwarding-options monitoring name output interface interface-name],
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output interface interface-name],
[edit forwarding-options sampling family (inet | inet6 | mpls) output interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the engine type number for flow monitoring and accounting services. The engine type attribute refers to the type of the flow switching engine, such as the route processor or a line module. The configured engine type is inserted in output **cflowd** packets. The **Source ID**, a 32-bit value to ensure uniqueness for all flows exported from a particular device, is the equivalent of the engine type and the engine ID fields.

> **NOTE:** You must configure a source address in the output interface statements. The interface-level statement of engine-type is added automatically but you can override this value with manually configured statements to track different flows with a single **cflowd** collector.

Options

number—Platform-specific accounting interface type.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Traffic Sampling on MX, M and T Series Routers | 375
- Configuring Flow Monitoring | 3
- Configuring Discard Accounting | 390

export-format

Syntax

```
export-format format;
```

Hierarchy Level

```
[edit forwarding-options monitoring name output]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Flow monitoring export format.

Options

format—Format of the flows.

Values: 5 or 8

Default: 5

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

version | **1213**

Exporting Flows | **6**

family (Monitoring)

Syntax

```
family inet {
    output {
        flow-active-timeout seconds;
        flow-inactive-timeout seconds;
        export-format format;
        cflowd hostname {
            aggregation {
                autonomous-system;
                destination-prefix;
                protocol-port;
                source-destination-prefix {
                    caida-compliant;
                }
                source-prefix;
            }
            port port-number;
        }
        interface interface-name {
            engine-id number;
            engine-type number;
            input-interface-index number;
            output-interface-index number;
            source-address address;
        }
    }
}
```

Hierarchy Level

```
[edit forwarding-options monitoring name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify input and output interfaces and properties for flow monitoring. Only IPv4 (**inet**) is supported.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Flow Monitoring | 3

family (Port Mirroring)

List of Syntax

MX, M, T Series Routers, EX Series Switches and SRX Series Firewalls on page 939
Syntax: QFX Series Switches, EX4600 and NFX Series Devices on page 939

MX, M, T Series Routers, EX Series Switches and SRX Series Firewalls

```
family (inet | inet6) {
    output {
        interface interface-name {
            next-hop address;
        }
        no-filter-check;
    }
}
```

Syntax: QFX Series Switches, EX4600 and NFX Series Devices

```
family
    ethernet-switching {
        output {
            interface interface-name {
            }
            no-filter-check;
            }
            vlan vlan-name {
                no-tag;
            }
        }
    inet
        output {
            ip-address address {
            }
            routing-instance instance-name {
                ip-address address {
            }
        }
```

Hierarchy Level

```
[edit forwarding-options port-mirroring]
```

Release Information

Statement introduced before Junos OS Release 7.4 for MX, M, T Series routers, EX Series switches and SRX Series firewalls.
Statement introduced in Junos OS Release 13.2 for the QFX Series and EX4600.

Description

Specify the type of interface that will be used to forward port mirrored packet to an analyzer device. Configure the protocol family to be sampled. Only IPv4 (**inet**) and IPv6 (**inet6**) are supported.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.
routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | **547**

Understanding Port Mirroring and Analyzers

Configuring Port Mirroring

Examples: Configuring Port Mirroring for Local Analysis

family (RFC2544 Benchmarking)

Syntax

```
family (bridge | ccc | inet);
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.
bridge option introduced in Junos OS Release 12.3X53 for ACX Series routers.
bridge option introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Configure the address type family for the benchmarking test.

Options

bridge—Run the test on a Layer 2 Ethernet line (E- Line) or an Ethernet LAN (E-LAN) service configured in a bridge domain. You can run the RFC2544-based benchmarking test only in the egress direction or the user-to-network interface (UNI) direction of an Ethernet line.

ccc—Run the test on a circuit cross-connect (CCC) or Ethernet pseudowire service. You can run the RFC2544-based benchmarking test either in the egress or ingress direction.

inet—Run the test on an IPv4 service.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- rfc2544-benchmarking | **1108**
- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

family (Sampling)

Syntax

```
family (inet | inet6 | mpls | vpls | bridge) {
    disable;
    input {
        max-packets-per-second max-packets-per-second;
        maximum-packet-length maximum-packet-length;
        rate rate;
        run-length run-length
        }
    output {
        aggregate-export-interval seconds;
        flow-active-timeout seconds;
        flow-inactive-timeout seconds;
        extension-service service-name;
        flow-server hostname {
            aggregation {
                autonomous-system;
                destination-prefix;
                protocol-port;
                source-destination-prefix {
                    caida-compliant;
                }
                source-prefix;
            }
            autonomous-system-type (origin | peer);
            dscp dscp-value;
            forwarding-class class-name;
            (local-dump | no-local-dump);
            port port-number;
            source-address address;
            version format;
            version9 {
                template template-name;
            }
            version-ipfix {
                template template-name;
            }
        }
        interface interface-name {
            engine-id number;
            engine-type number;
            source-address address;
```

```
    }
    file {
      disable;
      filename filename;
      files number;
      size bytes;
      (stamp | no-stamp);
      (world-readable | no-world-readable);
    }
    inline-jflow {
      source-address address;
      flow-export-rate rate;
    }
  }
}
```

Hierarchy Level

```
[edit forwarding-options sampling],
[edit forwarding-options sampling instance instance-name]
[edit forwarding-options sampling instance instance-name family (inet | inet6 | bridge)
```

Release Information

Statement introduced before Junos OS Release 7.4.
mpls option introduced in Release 8.3.
inet6 option introduced in Release 9.4.
vpls option added in Junos OS Release 13.2 for MX Series routers.
bridge option introduced in Release 18.2R1 for MX Series routers.

Description

Configure the protocol family to be sampled. IPv4 (**inet**) is supported for most purposes, but you can configure **family mpls** to collect and export MPLS label information, **family inet6** to collect and export IPv6 traffic using flow aggregation version 9, and **vpls** to collect and export VPLS information, and **bridge** to collect and export bridge information.

The remaining statements are explained separately. See CLI Explorer.

> **NOTE:** The **inline-jflow** statement is valid only under the **[edit forwarding-options sampling instance *instance-name* family inet output]** hierarchy level. The **file** statement is valid only under the **[edit forwarding-options sampling family inet output]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

file (Sampling)

Syntax

```
file {
    disable;
    filename filename;
    files number;
    size bytes;
    (stamp | no-stamp);
    (world-readable | no-world-readable);
}
```

Hierarchy Level

```
[edit forwarding-options sampling family inet output]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Collect the traffic samples in a file.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

file (Trace Options)

Syntax

```
file filename <files number <size bytes> <world-readable | no-world-readable>;
```

Hierarchy Level

```
[edit forwarding-options port-mirroring traceoptions],
[edit forwarding-options sampling traceoptions]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure information about the files that contain trace logging information.

Options

filename—Name of the file containing the trace information.

Default: /var/log/sampled

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Tracing Traffic Sampling Operations | 383

file-specification (File Format)

Syntax

```
file-specification {
    variant variant-number {
        data-format format;
        name-format format;
        transfer {
            record-level number;
            timeout seconds;
        }
    }
}
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure the file format for the flow collection files.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring File Formats | 272

file-specification (Interface Mapping)

Syntax

```
file-specification {
    variant variant-number;
}
```

Hierarchy Level

```
[edit services flow-collector interface-map]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the default file specification for interface mapping.

Options

variant-number—Default file format variant.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

filename

Syntax

```
filename filename;
```

Hierarchy Level

```
[edit forwarding-options sampling family (inet |inet6 |mpls) output file]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the name of the output file.

Options

filename—Name of the file in which to place the traffic samples. All files are placed in the directory **/var/tmp**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

filename-prefix

Syntax

```
filename-prefix prefix;
```

Hierarchy Level

```
[edit services flow-collector transfer-log-archive]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure the filename prefix for log files.

Options
prefix—Filename identifier.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

files

Syntax

```
files number;
```

Hierarchy Level

[edit forwarding-options **port-mirroring traceoptions file**],
[edit forwarding-options **sampling family** (inet |inet6 |mpls) **output file**],
[edit forwarding-options **sampling traceoptions file**]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the total number of files to be saved with samples or trace data.

Options

number—Maximum number of traffic sampling or trace log files. When a file named ***sampling-file*** reaches its maximum size, it is renamed ***sampling-file*.0**, then ***sampling-file*.1**, and so on, until the maximum number of traffic sampling files is reached. Then the oldest sampling file is overwritten.

Range: 1 through 100 files

Default: 5 files for sampling output; 10 files for trace log information

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

Configuring Traffic Sampling on MX, M and T Series Routers | 375

filter

Syntax

```
filter {
    input filter-name;
    output filter-name;
    group filter-group-number;
}
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number family inet]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Apply a firewall filter to an interface. You can also use filters for encrypted traffic.

Options

group *filter-group-number*—Use the specified interface to be part of a filter group. The default filter group number is 0.

input *filter-name*—Use the specified filter to evaluate when packets are received on the interface.

output *filter-name*—Use the specified filter to evaluate when packets are transmitted on the interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

> *Routing Policies, Firewall Filters, and Traffic Policers User Guide* or the *Junos OS Administration Library*
>
> Configuring Flow Monitoring | 3

flex-flow-sizing

Syntax

```
flex-flow-sizing;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services]
```

Release Information

Statement introduced in Junos OS Release 15.1F5 for MX Series routers.

NOTE: Workaround for **flex-flow-sizing** on MX204 router:

```
Replace flex-flow-sizing with below configuration and reload the box.
Configure flow-table-size within a range of 1 through 15. If flex-flow-sizing
 is configured, deactivate or delete the same.
For example:
flow-table-size {
    ipv4-flow-table-size 10;
    ipv6-flow-table-size 4;
}
```

Description

Configure support for the service creation of flows for inline services sampling. This configuration results in a first-come-first-serve creation of flows. Whichever flow comes first, that is allowed to occupy the flow-table if there is space in the table. Otherwise, the flow is dropped and an error count is created.

NOTE: You cannot configure the **explicit flow-table-sizes** because **flex-flow-sizing** and **explicit flow-table-sizes** are mutually exclusive.

You need not perform **fpc** reboot to change from **flex** to **per family** configuration.

Options

Default: 1K flows for IPv6 and VPLS flows each.

Range: 15 through 256K flows for IPv4.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | **63**

Including Fragmentation Identifier and IPv6 Extension Header Elements in IPFIX Templates on MX Series Routers | **536**

flow-active-timeout

Syntax

```
flow-active-timeout seconds;
```

Hierarchy Level

```
[edit forwarding-options accounting name output],
[edit forwarding-options monitoring name output],
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls | vpls) output],
[edit forwarding-options sampling family (inet |inet6 |mpls |vpls) output],
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name],
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level added in Junos OS Release 10.2.

Support at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level added in Junos OS Release 16.1 for MPLS traffic flows.

Description

Set the interval after which an active flow is exported.

> **NOTE:** The router must include an Adaptive Services, Multiservices, or Monitoring Services PIC for this statement to take effect.

Options

seconds—Duration of the timeout period.

Range: 60 through 1800 seconds (for **forwarding-options** configurations); 10 through 600 seconds (for **services** configurations)

Default: 1800 seconds (for **forwarding-options** configurations); 60 seconds (for **services** configurations)

NOTE: In active flow monitoring, the cflowd or flow monitoring version 9 records are exported after a time period that is a multiple of 60 seconds and greater than or equal to the configured active timeout value. For example, if the active timeout value is 90 seconds, the cflowd or flow monitoring version 9 records are exported at 120-second intervals. If the active timeout value is 150 seconds, the cflowd or flow monitoring version 9 records are exported at 180-second intervals, and so forth.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Time Periods When Flow Monitoring Is Active and Inactive | **7**
- Configuring the Version 9 Template Properties | **496**
- Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

flow-collector

Syntax

```
flow-collector {
    analyzer-address address;
    analyzer-id name;
    destinations {
        ftp:url {
            password "password";
        }
    }
    file-specification {
        variant variant-number {
            data-format format;
            name-format format;
            transfer {
                record-level number;
                timeout seconds;
            }
        }
    }
    interface-map {
        collector interface-name;
        file-specification variant-number;
        interface-name {
            collector interface-name;
            file-specification variant-number;
        }
    }
    retry number;
    retry-delay seconds;
    transfer-log-archive {
        archive-sites {
            ftp:url {
                password "password";
                username username;
            }
        }
        filename-prefix prefix;
        maximum-age minutes;
    }
}
```

Hierarchy Level

[edit services]

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Define the flow collection.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Flow Collection Overview | 269

flow-control-options

Syntax

```
flow-control-options {
    down-on-flow-control;
    dump-on-flow-control;
    reset-on-flow-control;
    up-on-flow-control;
}
```

Hierarchy Level

[edit **interfaces** mo-*fpc/pic/port* **multiservice-options**]

Release Information

Statement introduced before Junos OS Release 8.4.

Description

Configure the flow control options for application recovery in case of a prolonged flow control failure.

- **down-on-flow-control**—Bring interface down during prolonged flow control.
- **dump-on-flow-control**—Cause core dump during prolonged flow control.

  NOTE: Starting with Junos OS Release 15.1, on MX Series routers with MS-MICs and MS-MPCs, instead of an eJunos kernel core file, the multiservices PIC management daemon (mspmand) core file is generated when a prolonged flow control failure occurs and when you configure the setting to generate a core dump during prolonged flow control (by using the **dump-on-flow-control** option with the **flow-control-options** statement). The watchdog functionality continues to generate a kernel core file in such scenarios.

- **reset-on-flow-control**—Reset interface during prolonged flow control.

  NOTE: Starting in Junos OS Release 16.1R7, the **reset-on-flow-control** option has no effect on the MS-MIC, MS-MPC, MS-DPC, MS-PIC 100, MS-PIC 400, and MS-PIC 500 line cards. This is because starting in Release 16.1R7, Junos OS restarts these line cards to recover them from stuck state due to prolonged flow control.

- **up-on-flow-control**—Cause interface to remain in stuck state until you manually restart the PICs.

NOTE: Starting in Junos OS Release 16.1R7, if interfaces on an MS-PIC or MS-DPC are in stuck state because of prolonged flow control, Junos OS restarts the service PICs to recover them from this state. However, if you want the PICs to remain in stuck state until you manually restart the PICs, configure the **up-on-flow-control** option. In releases before Release 16.1R7, there is no action taken to recover service PICs from this state unless one of the options for the **flow-control-options** statement is configured, or service PIC is manually restarted.

Usage Guidelines

See "Configuring Flow Monitoring" on page 3.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

flow-export-destination

Syntax

```
flow-export-destination {
    (cflowd-collector | collector-pic);
}
```

Hierarchy Level

[edit forwarding-options **monitoring** *group-name* family inet **output**]

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure flow collection.

Options
cflowd-collector—Use the cflowd collector.

collector-pic—Use the collector PIC.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Exporting Flows | 6

flow-export-rate

Syntax

```
flow-export-rate rate;
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family inet output inline-jflow]
```

Release Information

Statement introduced in Junos OS Release 10.2.

Description

Specify the flow export rate of monitored packets in kpps. If you have multiple line cards of different types running on the same router, the **flow-export-rate** will be applied to each card. However, the rate applied to the PFEs on the card will vary in accordance with the number of PFEs that are on the card.

On the MX Series, the actual flow export rate might be less than the configured **flow-export-rate**. In addition, the maximum flow export rate for an FPC or MPC module cannot exceed the configured **flow-export-rate**, regardless of how many PICs or MICs are on the module.

Options

rate—Flow export rate of monitored packets in kpps (from 1 through 400).

Default: 1 kpps (applies to all PFEs on the FPC)

> **NOTE:** The maximum rate per PFE is 100 kpps for LU, 800 kpps for XL/EA, so for an FPC with four LU PFEs (such as AS cards) you can set a maximum **flow-export-rate** of 400. For an FPC with two LU PFEs (such as the MPC2), the maximum **flow-export-rate** is 200. For an FPC with one LU PFE (such as the MPC5), the maximum is 100. The Junos CLI accepts as valid any value within the range of 1 to 3200, but when applied the value might trigger an error message such as “The configured flow export rate is higher than supported value/chip” in the Junos message log.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Discard Accounting | 390

Configuring Flow Monitoring | 3

Configuring Traffic Sampling on MX, M and T Series Routers | 375

flow-inactive-timeout

Syntax

```
flow-inactive-timeout seconds;
```

Hierarchy Level

```
[edit forwarding-options accounting name output],
[edit forwarding-options monitoring name output],
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls | vpls) output],
[edit forwarding-options sampling family (inet |inet6 |mpls) output],
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name],
```

Release Information

Statement introduced before Junos OS Release 7.4.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level added in Junos OS Release 10.2.
Support at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level added in Junos OS Release 16.1 for MPLS traffic flows.

Description

Set the interval of inactivity that marks a flow inactive.

> **NOTE:** The router must include an Adaptive Services, Multiservices, or Monitoring Services PIC for this statement to take effect.

Options

seconds—Duration of the timeout period.

Range: 15 through 1800 seconds (for **forwarding-options** configurations); 10 through 600 seconds (for **services** configurations)

Default: 60 seconds (for **forwarding-options** configurations); 60 seconds (for **services** configurations)

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Time Periods When Flow Monitoring Is Active and Inactive | 7

Configuring the Version 9 Template Properties | 496

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | 513

flow-key (Flow Monitoring)

Syntax

```
flow-key {
    flow-direction;
    vlan-id;
    output-interface;
}
```

Hierarchy Level

[edit services flow-monitoring version9 template *template-name*],
[edit services flow-monitoring version-ipfix template *template-name*]

Release Information

Statement introduced in Junos OS Release 15.2.
output-interface option added in Junos OS Release 18.2R1.

Description

Include VLAN IDs in both the ingress and egress directions in the flow key, enable flow direction information in a Version 9 or IPFIX flow template, or both, and configure the output-interface for bridge or VPLS family for inline flow monitoring on the MX Series.

Options

flow-direction—Enable reporting of the direction of the flow. The field contains 0x00 (ingress) or 0x01 (egress). The flow direction field in the output record contains the invalid value 0xFF if you do not configure **flow-direction**.

vlan-id—Include VLAN IDs in both the ingress and egress directions in the flow key.

output-interface—Configure the output-interface field as part of flow-key for bridge or VPLS family.

NOTE: If the output-interface (OIF) is configured under flow-key while the flow-monitoring is in progress, all the existing flows (where OIF was not part of flow-key) report OIF field as zero in the next export. Therefore, in progress configuration of output-interface as part of flow-key is not recommended. In order to configure output-interface as part of flow-key, it is recommended to disable the bridge or vpls sampling and wait for the active flows to become zero.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | **63**
- Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**
- Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

flow-monitoring

Syntax

```
flow-monitoring {
  version9 {
    template template-name {
      options-template-id
      template-id
      source-id
      flow-active-timeout seconds;
      flow-inactive-timeout seconds;
      flow-key {
        flow-direction;
        vlan-id;
        output-interface;
      }
      (ipv4-template | ipv6-template | mpls-template label-position [ positions ] | mpls-ipv4-template label-position
        [ positions ] | mpls-ipvx-template);
      peer-as-billing-template;
      option-refresh-rate packets packets seconds seconds;
      options-template-id
      source-id
      template-id
      template-refresh-rate packets packets seconds seconds;
      tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
    }
  }
  version-ipfix {
    template template-name {
      flow-active-timeout seconds;
      flow-inactive-timeout seconds;
      flow-key {
        flow-direction;
        vlan-id;
      }
      (ipv4-template | ipv6-template | mpls-ipv4-template | mpls-ipvx-template | vpls-template);
      nexthop-learning (enable |disable);
      observation-domain-id
      option-refresh-rate packets packets seconds seconds;
      options-template-id
      template-id
      template-refresh-rate packets packets seconds seconds;
      tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
    }
```

```
    }
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Specify the active monitoring properties for flow aggregation version 9 or IPFIX.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**
- Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

flow-server

Syntax

```
flow-server hostname {
    aggregation {
        autonomous-system;
        destination-prefix;
        protocol-port;
        source-destination-prefix {
            caida-compliant;
        }
        source-prefix;
    }
    autonomous-system-type (origin | peer);
    dscp dscp-value;
    forwarding-class class-name;
    (local-dump | no-local-dump);
    port port-number;
    source-address address;
    version format;
    version9 {
        template template-name;
    }
}
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls | vpls | bridge) output],
[edit forwarding-options sampling family (inet | inet6 | mpls | vpls | bridge) output]
```

Release Information

Statement introduced before Junos OS Release 7.4.
version9 statement introduced in Junos OS Release 8.3.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Support at the following hierarchy levels introduced in Junos OS Release 18.2R1: **[edit forwarding-options sampling instance instance-name family bridge]**, , **[edit forwarding-options sampling family bridge]**.

Description

Collect an aggregate of sampled flows and send the aggregate to a specified host system that runs the collection utility cfdcollect. Specify a host system to collect sampled flows using the version 9 format.

You can configure up to one version 5 and one version 8 flow format at the **[edit forwarding-options sampling family (inet | inet6 | mpls) output flow-server *hostname*]** hierarchy level. For the same configuration,

you can specify only either version 9 flow record formats or formats using versions 5 and 8, not both types of formats.

Options

hostname—IP address–IPv4 or IPv6–or identifier of the host system (the workstation either running the cflowd utility or collecting traffic flows using version 9).

> **NOTE:** Only host systems running IPv4 are supported on QFX10000 switches.

You can configure only one host system for version 9.

> **NOTE:** IPv6 coniguration for **flow-server** is supported only in Junos OS Release 12.3 and later.
>
> Note that when you configure an IPv6 address for the **flow-server** statement, you must also configure an IPv6 address for the **inline-jflow source-address** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6 | mpls | vpls | bridge) output]** hierarchy level.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

flow-table-size

Syntax

```
flow-table-size {
    ipv4-flow-table-size units;
    ipv6-extended-attrib;
    ipv6-flow-table-size units;
    mpls-flow-table-size units;
    vpls-flow-table-size units;
}
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services]
```

Release Information

Statement introduced in Junos OS Release 12.1.
ipv6-extended-attrib option added in Junos OS Release 14.2 for MX Series routers.
vpls-flow-table-size option added in Junos OS Release 13.2 for MX Series routers.
bridge-flow-table-size option added in Junos OS Release 18.2R1 for MX Series routers.

Description

Configure the size of hash tables for inline services sampling.

Starting with Junos OS Release 15.1F2, by default, the software allocates one 1K IPv4 flow table. To allocate 15 256K IPv4 flow tables, the former default, you can enter this configuration from the **[edit]** hierarchy level:

```
[edit]
user@router# set chassis fpc inline-services flow-table-size ipv4-flow-table-size
15
```

NOTE: The recommended flow table size is 4 so that it can scale up to 4x256K flows, which is 1M. You can configure more, however, the system will issue a warning message.

> **NOTE:** Starting from Junos OS Release 16.1R1 and 15.1F2, any changes in the configured size of the flow table needs a reboot of the FPC. We recommend that you run this command during a maintenance window. The maximum supported flow table size for a combination of both IPv4 and IPv6 is 15. For example, you can set the flow table size for IPv4 to 10 and set the size for IPv6 to 5. Verify that you have sized the hash tables adequately for IPv4 and IPv6 flow sampling.

The remaining statements are defined separately.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | **63**

Including Fragmentation Identifier and IPv6 Extension Header Elements in IPFIX Templates on MX Series Routers | **536**

flow-table-size (Chassis)

Syntax

```
flow-table-size size;
```

Hierarchy Level

```
[edit chassis fpc slot inline-video-monitoring]
```

Release Information

Statement introduced in Junos OS Release 16.1 on the MX Series.

Description

Configure the number of video flows that can be measured per Packet Forwarding Engine by an MPC at a given time. This value takes effect the next time the MPC is rebooted.

Options

size—Number of video flows per Packet Forwarding Engine.

Range: 16 through 8192

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Video Monitoring on MX Series Routers | 833

flow-tap

Syntax

```
flow-tap {
    (interface interface-name | tunnel-interface interface-name);
    family (inet | inet6);
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 8.1.
ccc option introduced in Junos OS Release 17.2.

Description

Enable the flow-tap service or FlowTapLite service on an interface. FlowTapLite is a lighter version of the flow-tap application that is available only on tunnel interfaces on MX Series platforms, M120 Series routers, and M320 Series routers with Enhanced III FPCs only.

Starting in Junos OS Release 17.3R1, the FlowTapLite service can run concurrently with the radius-flow-tap service on the same MX Series router. The radius-flow-tap service (**[edit services radius-flow-tap]**) is required for subscriber secure policy mirroring on MX Series routers.

In earlier releases, the FlowTapLite and radius-flow-tap services cannot run concurrently on an MX Series router, which prevents you from running FlowTapLite monitoring and subscriber secure policy mirroring at the same time.

Options

interface *interface-name*—Use the specified interface for the flow-tap application.

tunnel-interface *interface-name*—Use the specified tunnel interface for the FlowTapLite application.

family—(Not applicable for FlowTapLite) Apply flow-tap services to the specified family. If you do not specify an option, the flow-tap service is applied only to IPv4 traffic.

- inet—IPv4 traffic.
- inet6—IPv6 traffic.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Release History Table

Release	Description
17.3R1	Starting in Junos OS Release 17.3R1, the FlowTapLite service can run concurrently with the radius-flow-tap service on the same MX Series router.

RELATED DOCUMENTATION

Configuring Junos Packet Vision on MX, M and T Series Routers | 335

forwarding-class (Sampling)

Syntax

```
forwarding-class class-name;
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet | inet6) output flow-server hostname]
```

Release Information

Statement introduced in Junos OS Release 16.1 on the MX Series and the PTX Series.

Description

Specify the forwarding class to which exported packets for inline active flow monitoring are sent.

Default

If you do not include the ***forwarding-class*** statement, exported packets are sent to the best effort queue.

Options

forwarding-class *class-name*—Name of the forwarding class:

- **assured-forwarding**
- **best-effort**
- **expedited-forwarding**
- **network-control**

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Sampling Instance on MX, M and T Series Routers or QFX Series Switches | 388

ftp (Flow Collector Files)

Syntax

```
ftp:url;
```

Hierarchy Level

```
[edit services flow-collector destination]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Specify the primary and secondary destination FTP server addresses.

Options
url—FTP server address. The URL can include the following macros, typed in braces:

- **{%D}**—Date
- **{%T)**—Time when the file is created
- **{%I}—**Description string for the logical interface configured using the **collector *interface-name*** statement at the **[edit services flow-collector interface-map]** hierarchy
- **{%N}**—Unique, sequential number for each new file created
- **{am_pm}**—**AM** or **PM**
- **{date}**—Current date using the **{year} {month} {day}** macros
- **{day}**—From **01** through **31**
- **{day_abbr}**—**Sun** through **Sat**
- **{day_full}**—**Sunday** through **Saturday**
- **{generation number}**—Unique, sequential number for each new file created
- **{hour_12}**—From **01** through **12**
- **{hour_24}**—From **00** through **23**
- **{ifalias}**—Description string for the logical interface configured using the **collector** statement at the **[edit services flow-collector interface-map]** hierarchy
- **{minute}**—From **00** through **59**
- **{month}**—From **01** through **12**

- **{month_abbr}**—**Jan** through **Dec**
- **{month_full}**—**January** through **December**
- **{num_zone}**—From **-2359** to **+2359**; this macro is not supported
- **{second}**—From **00** through **60**
- **{time}**—Time the file is created, using the **{hour_24} {minute} {second}** macros
- **{time_zone}**—Time zone code name of the locale; for example, **gmt** (this macro is not supported).
- **{year}**—In the format YYYY; for example, **1970**
- **{year_abbr}**—From **00** through **99**

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Destination FTP Servers for Flow Records | 271

ftp (Transfer Log Files)

Syntax

```
ftp:url;
```

Hierarchy Level

```
[edit services flow-collector transfer-log-archive archive-sites]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the primary and secondary destination FTP server addresses.

Options

url—FTP server address.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

g-duplicates-dropped-periodicity

Syntax

```
g-duplicates-dropped-periodicity seconds;
```

Hierarchy Level

```
[edit services dynamic-flow-capture]
```

Release Information
Statement introduced in Junos OS Release 9.2.

Description
Specify the frequency for sending notifications to affected control sources when transmission of duplicate sets of data is restricted because the **g-max-duplicates** threshold has been reached. This setting is applied globally; the **duplicates-dropped-periodicity** setting applied at the **capture-group** level overrides the global setting.

Default
The default period for sending notifications is 30 seconds.

Options
seconds—Period for sending DuplicatesDropped notifications.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

duplicates-dropped-periodicity | 931
Limiting the Number of Duplicates of a Packet | 328

g-max-duplicates

Syntax

```
g-max-duplicates number;
```

Hierarchy Level

```
[edit services dynamic-flow-capture]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify the maximum number of content destinations to which DFC PICs can send data from a single input set of packets. Limiting the number of duplicates reduces the load on the PIC. This setting is applied globally; the **max-duplicates** setting applied at the **capture-group** level overrides the global setting.

Default

If no value is configured, a default setting of 3 is used.

Options

number—Maximum number of content destinations.

Range: 1 through 64

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

max-duplicates | **1027**

Limiting the Number of Duplicates of a Packet | **328**

generate-snmp-traps

Syntax

```
generate-snmp-traps;
```

Hierarchy Level

```
[edit services]
[edit services video-monitoring]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

If this statement is configured, the service generates SNMP traps for severity levels such as Info, Warning, Critical, or Cleared. For example, if DF alarm changes from info to warning, or from warning to critical, mdiDFAlarm trap be triggered.

> **NOTE:** SNMP traps are not generated if SNMP trap generation is not enabled.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

alarms | 869

hard-limit

Syntax

```
hard-limit bandwidth;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name content-destination identifier]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify a bandwidth threshold at which the dynamic flow capture application begins deleting criteria, until the bandwidth falls below the **hard-limit-target** value.

Options

bandwidth—Hard limit threshold, in bits per second.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

hard-limit-target | 985
Configuring the Content Destination | 323

hard-limit-target

Syntax

```
hard-limit-target bandwidth;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name content-destination identifier]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify a bandwidth threshold at which the dynamic flow capture application stops deleting criteria.

Options

bandwidth—Target value, in bits per second.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

hard-limit | 984

Configuring the Content Destination | 323

hardware-timestamp

Syntax

```
hardware-timestamp;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name]
```

Release Information

Statement introduced in Junos OS Release 8.1.
Statement applied to MX Series routers in Junos OS Release 10.0.
Statement introduced in Junos OS Release 10.3 for EX Series switches.
Statement introduced in Junos OS Release 19.1 for PTX Series routers.
Statement introduced in Junos OS Release 19.2R1 for MPC10E-15C-MRATE on MX240, MX480, and MX960 routers.
Statement introduced in Junos OS Release 19.2R1 for MPC11E on MX2008, MX2010, and MX2020 routers.

Description

Enable timestamping of RPM probe messages in the Packet Forwarding Engine host processor. This feature is supported only with **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, and **udp-ping-timestamp** probe types.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

history-size

Syntax

```
history-size size;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Statement at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the number of stored history entries.

Options

size—Value from 0 to 255.

Default: 50

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | **629**

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

host-outbound media-interface

Syntax

```
host-outbound media-interface;
```

Hierarchy Level

```
[edit chassis]
```

Release Information
Statement introduced in Junos OS Release 13.2 on MX Series 5G Universal Routing Platforms.

Description
Enable Layer 2 port mirroring of host-generated outbound packets only on MPCs on MX Series 5G Universal Routing Platforms.

This statement enables all Routing Engine-generated Layer 2 injections to execute egress logical interface filters.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- *Examples: Layer 2 Port Mirroring at Multiple Levels of the Chassis*
- *Configuring Port Mirroring*
- *Understanding Layer 2 Port Mirroring*

in-service (RFC2544 Benchmarking)

Syntax

```
in-service;
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Runs the test in the **in-service** mode. In this mode, while the test is running, the rest of the data traffic sent to and from the UNI port under test on the service are not interrupted. Control protocol packets and control protocol peering are not interrupted.

If this mode is not configured, the test runs in the default **out-of-service** mode. In the **out-of-service** mode, while the test is running, all the data traffic sent to and from the UNI port under test on the service is interrupted. Control protocol peering is not interrupted whereas control protocol packets such as CFM sessions are interrupted.

Default

The default service mode for the reflecting egress interface for an E-LAN service is **out-of-service** mode.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

rfc2544-benchmarking | **1108**

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

Configuring an RFC 2544-Based Benchmarking Test | **659**

inactivity-timeout (Services RPM)

Syntax

```
inactivity-timeout seconds;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Inactivity timeout period, in seconds.

Options

seconds—Length of time the session is inactive before it times out.

Default: 1800 seconds

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

inline-jflow

Syntax

```
inline-jflow {
    source-address address;
    flow-export-rate rate;
}
```

Hierarchy Level

[edit forwarding-options **sampling instance** *instance-name* **family** inet **output**]

Release Information

Statement introduced in Junos OS Release 10.2.
Statement introduced in Junos OS Release 14.2 for T4000 Series routers with Type 5 FPC.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Specify inline flow monitoring for traffic from the designated address.

The remaining statements are explained separately. See CLI Explorer.

> **NOTE:** If you configure inline flow monitoring with **inline-jflow** then you have to disable it before performing ISSU. For more information, see *Before You Begin a Unified ISSU*.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

inline-monitoring

Syntax

```
inline-monitoring {
    instance instance-name;
    template template-name;
    traceoptions;
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Configure inline monitoring service. When you enable inline monitoring, you can monitor actual IPv4 and IPv6 packets at different sampling rates, and export the actual packet up to the configured clip length. By default, Junos OS supports a maximum packet length of 126 byte starting with the Ethernet header. The monitored packets are exported to an collector for further processing. The packets are exported in an IPFIX format, which includes information on the original packet size, and incoming or outgoing interface.

Options

instance *instance-name*—Configure an inline-monitoring instance parameters. See **instance** for more information.

template *template-name*—Configure templates for inline packet monitoring. See **template** for more information.

traceoptions—(Optional) Configure traceoptions for the inline monitoring process. See **traceoptions** for more information.

Required Privilege Level

system

RELATED DOCUMENTATION

Understanding Inline Monitoring Services | 362

instance

Syntax

```
instance name {
    collector name;
    maximum-clip-length maximum-clip-length;
    template-name template-name;
}
```

Hierarchy Level

```
[edit services inline-monitoring]
```

Release Information

Statement introduced in Junos OS 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Configure inline-monitoring instance parameters. You can use these instances along with firewall filters to monitor different streams of traffic at different sampling rates from the same interface.

You can configure a maximum of sixteen inline-monitoring instances with four collectors each.

Options

name—Name of instance.

collector *name*—Configure an collector for the inline-monitoring instance. See **collector** for more information.

maximum-clip-length—Maximum packet length.

Range: 64 through 126 bytes

Default: 126 bytes

template-name—Name of the template.

Required Privilege Level

system

RELATED DOCUMENTATION

Understanding Inline Monitoring Services | 362

input (Port Mirroring)

Syntax

```
input {
    maximum-packet-lengthbytes
    rate number;
    run-length number;
}
```

Hierarchy Level

```
[edit forwarding-options port-mirroring],
[edit forwarding-options port-mirroring instance instance-name],
[edit forwarding-options port-mirroring family (inet | inet6)]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure port mirroring on a logical interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

input (Sampling)

Syntax

```
input {
    max-packets-per-second number;
    rate number;
    run-length number;
    maximum-packet-length bytes;
}
```

Hierarchy Level

```
[edit forwarding-options sampling],
[edit forwarding-options sampling instance instance-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure traffic sampling on a logical interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

input-interface-index

Syntax

```
input-interface-index number;
```

Hierarchy Level

```
[edit forwarding-options monitoring name output interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify a value for the input interface index that overrides the default supplied by SNMP.

Options

number—Input interface index value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

input-packet-rate-threshold

Syntax

```
input-packet-rate-threshold rate;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Specify a packet rate threshold value that triggers a system log warning message.

Options

rate—Threshold value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Thresholds | 328

instance (Sampling)

Syntax

```
instance instance-name {
    disable;
    family (bridge | inet | inet6 | mpls | vpls) {
        disable;
        output {
            aggregate-export-interval seconds;
            flow-active-timeout seconds;
            flow-inactive-timeout seconds;
            extension-service service-name;
            flow-server hostname {
                aggregation {
                    autonomous-system;
                    destination-prefix;
                    protocol-port;
                    source-destination-prefix {
                        caida-compliant;
                    }
                    source-prefix;
                }
                autonomous-system-type (origin | peer);
                dscp dscp-value;
                forwarding-class class-name;
                (local-dump | no-local-dump);
                port port-number;
                source-address address;
                version format;
                version9 {
                    template template-name;
                }
                version-ipfix {
                    template template-name;
                }
            }
            interface interface-name {
                engine-id number;
                engine-type number;
                source-address address;
            }
            inline-jflow {
                source-address address;
                flow-export-rate rate;
```

```
        }
      }
    }
    input {
      rate number;
      run-length number;
      max-packets-per-second number;
      maximum-packet-length bytes;
    }
}
```

Hierarchy Level

```
[edit forwarding-options sampling]
```

Release Information

Statement introduced in Junos OS Release 9.6.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Configure a sampling instance.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Sampling Instance on MX, M and T Series Routers or QFX Series Switches | 388

interface (Accounting or Sampling)

Syntax

```
interface interface-name {
    engine-id number;
    engine-type number;
    source-address address;
}
```

Hierarchy Level

```
[edit forwarding-options accounting name output],
[edit forwarding-options sampling family (inet | inet6 | mpls) output],
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls) output]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the output interface for monitored traffic.

Options

interface-name—Name of the interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Discard Accounting | 390

Configuring Traffic Sampling on MX, M and T Series Routers | 375

Understanding IP-Based Filtering and Selective Port Mirroring of MPLS Traffic

interfaces

Syntax

```
interfaces { ... }
```

Hierarchy Level

```
[edit]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure interfaces on the router.

Default
The management and internal Ethernet interfaces are automatically configured. You must configure all other interfaces.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Junos OS Network Interfaces Library for Routing Devices

interface (Services Flow Tap)

Syntax

```
interface sp-fpc/pic/port.logical-unit-number;
```

Hierarchy Level

```
[edit services flow-tap]
```

Release Information
Statement introduced in Junos OS Release 8.1.

Description
Specify the AS PIC interface used with the flow-tap application. Any AS PIC available in the router can be assigned, and any logical interface on the AS PIC can be used.

Options
sp-*fpc/pic/port.logical-unit-number*—Use the specified services interface for flow-tap service.

You cannot configure flow-tap services on channelized interfaces.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Junos Packet Vision Interface | 335

interface-map

Syntax

```
interface-map {
    collector interface-name;
    file-specification variant-number;
    interface-name {
        collector interface-name;
        file-specification variant-number;
    }
}
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Match an input interface with a flow collector interface and apply the preset file specifications to the input interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Interface Mappings | 273

interfaces (Services Dynamic Flow Capture)

Syntax

```
interfaces interface-name;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Specify the DFC interface used with the control source configured in the same capture group.

Options

interface-name—Name of the DFC interface.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the DFC PIC Interface | 325

interfaces (Video Monitoring)

Syntax

```
interfaces {
    interface-name {
        family {
            inet {
                input-flows {
                    input-flow-name {
                        destination-address [ address ];
                        destination-port [ port ];
                        source-address [ address ];
                        source-port [ port ];
                        template template-name;
                    }
                }
                output-flows {
                    output-flow-name {
                        destination-address [ address ];
                        destination-port [ port ];
                        source-address [ address ];
                        source-port [ port ];
                        template template-name;
                    }
                }
            }
            inet6 {
                input-flows {
                    input-flow-name {
                        destination-address [ address ];
                        destination-port [ port ];
                        source-address [ address ];
                        source-port [ port ];
                        template template-name;
                    }
                }
                output-flows {
                    output-flow-name {
                        destination-address [ address ];
                        destination-port [ port ];
                        source-address [ address ];
                        source-port [ port ];
                        template template-name;
                    }
```

```
            }
        }
        mpls {
            input-flows {
                input-flow-name {
                    (destination-address [ address ] | source-address [ address ]);
                    destination-port [ port ];
                    payload-type (ipv4 | ipv6);
                    source-port [ port ];
                    template template-name;
                }
            }
            output-flows {
                output-flow-name {
                    (destination-address [ address ] | source-address [ address ]);
                    destination-port [ port ];
                    payload-type (ipv4 | ipv6);
                    source-port [ port ];
                    template template-name;
                }
            }
        }
    }
  }
}
```

Hierarchy Level

[edit services **video-monitoring**]

Release Information

Statement introduced in Junos OS Release 14.1.
mpls option introduced in Junos OS Release 17.2.
payload-type ipv6 option introduced in Junos OS Release 17.4.

Description

Define video monitoring for specified input or output flows on selected interfaces. You can configure a maximum of 256 flows for an interface.

Options

destination-address *address*—Destination IPv4 or IPv6 address or prefix value of a flow that you want to monitor. You can use up to 32 addresses.

For IPv4-over-MPLS flows, if you configure multiple addresses for both the destination and source, then either all the destination or all the source values must have the same prefix length. For example, the following is allowed, because all the destination addresses have the same prefix length.

```
[edit services video-monitoring interfaces ge-0/2/2.0 family mpls]
user@host# set input-flows input-flow-name destination-address [203.0.13.0/24 198.51.100.0/24]
user@host# set input-flows input-flow-name source-address [172.16.0.0/12 192.0.2.11/32]
```

For IPv6-over-MPLS flows, if you configure both the destination and source address, you can use multiple addresses for either the destination or the source IP address, but not for both.

destination-port *port*—Destination port number of a flow that you want to monitor. You can use multiple port numbers and port ranges.

Range: 0 through 65,535

input-flows *input-flow-name*—Name of an input flow you are defining.

interface-name—Name of the interface to monitor.

output-flows *output-flow-name*—Name of an output flow you are defining.

payload-type ipv4—Monitor video stream for IPv4-over-MPLS traffic.

payload-type ipv6—Monitor video stream for IPv6-over-MPLS traffic.

source-address *address*—Source IPv4 or IPv6 address or prefix value of a flow that you want to monitor. You can use up to 32 addresses.

For IPv4-over-MPLS flows, if you configure multiple addresses for both the destination and source, then either all the destination or all the source values must have the same prefix length. For example, the following is allowed, because all the destination addresses have the same prefix length.

```
[edit services video-monitoring interfaces ge-0/2/2.0 family mpls]
user@host# set input-flows input-flow-name destination-address [203.0.13.0/24 198.51.100.0/24]
user@host# set input-flows input-flow-name source-address [172.16.0.0/12 192.0.2.11/32]
```

For IPv6-over-MPLS flows, if you configure both the destination and source address, you can use multiple addresses for either the destination or the source IP address, but not for both.

source-port *port*—Source port number of a flow that you want to monitor. You can use multiple port numbers and port ranges.

Range: 0 through 65,535

template-name—Name of the template used to monitor the input flows or output flows on an interface.

The template contains the measurement parameters for video monitoring, and is configured at the **[edit services video-monitoring templates]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Video Monitoring on MX Series Routers | 833

inet6-options (Services)

Syntax

```
inet6-options {
    source-address address;
}
```

Hierarchy Level

```
[edit services rpm probe owner test test-name]
```

Release Information

Statement introduced in Junos OS Release 14.1R4.

Description

Specify the source IPv6 address used for probes. If the source IPv6 address is not one of the devices' assigned addresses, the packet uses the outgoing interface's address as its source.

Options

inet6-options—Use the specified base IPv6 protocol-related settings to be used for RPM probes

source-address *ipv6-address*—Specify the base IPv6 address for sending the RPM probes from the client to the server (for example, 2001:db8::a:b:c:d).

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

ip-swap (RFC 2544 Benchmarking)

Syntax

```
ip-swap;
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX Series routers.

Description

Swaps source and destination IPv4 addresses. This statement is applicable only for family **bridge**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- Configuring an RFC 2544-Based Benchmarking Test | **659**
- rfc2544-benchmarking | **1108**

ipv4-flow-table-size

Syntax

```
ipv4-flow-table-size units;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services flow-table-size]
```

Description

Configure the size of the IPv4 flow table in units of 256K entries.

> **NOTE:** Prior to Junos OS Release 16.1R1 and 15.1F2, any changes in the configured size of the flow table initiates an automatic reboot of the FPC, and we recommend that you run this command in a maintenance window.

> **NOTE:** The recommended flow table size is 4 so that it can scale up to 4x256K flows, which is 1M. You can configure more, however, the system will issue a warning message.

Starting with Junos OS Release 16.1R1 and 15.1F2, by default, the software allocates 1K entries for IPv4 flow tables. To allocate fifteen 256K IPv4 flow tables, the former default, you can enter this configuration from the **[edit]** hierarchy level:

```
[edit]
user@router# set chassis fpc slot-number inline-services flow-table-size
ipv4-flow-table-size 15
```

Starting with Junos OS Release 17.3R1, for LU-based platforms, the maximum number of units is 15. For XL-based platforms, the maximum is 220. For EA-based platforms, the maximum is 48 for MPC7E and MPC9E. For MPC8E, the maximum is 97.

Options

units—Number of 256K flow entries available for the IPv4 flow table.

Range: 1 through 245

Default: 1024 (1K)—Starting with Junos OS Release 16.1R1 and 15.1F2

Default: 3,932,160 (3840K)—Prior to Junos OS Release 16.1R1 and 15.1F2

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

Release History Table

Release	Description
17.3R1	Starting with Junos OS Release 17.3R1, for LU-based platforms, the maximum number of units is 15. For XL-based platforms, the maximum is 220. For EA-based platforms, the maximum is 48 for MPC7E and MPC9E. For MPC8E, the maximum is 97.
16.1R1	Starting with Junos OS Release 16.1R1 and 15.1F2, by default, the software allocates 1K entries for IPv4 flow tables.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

ipv4-template

Syntax

```
ipv4-template;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced in Junos OS Release 8.3.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level added in Junos OS Release 10.2.
Statement introduced in Junos OS Release 15.1F4 for PTX Series routers with third-generation FPCs installed. Supported at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.

Description

Specify the version 9 or IPFIX template properties for one of the following:

- Template for monitoring IPv4 flows.
- Template for inline monitoring an MPLS-over-UDP flow that is carried between IPv4 endpoints on PTX Series routers. This monitoring looks past the tunnel header to report the inner payload of the packets. To use the template for MPLS-over-UDP flows, you must also configure **tunnel-observation mpls-over-udp** at the **[edit services flow-monitoring (version 9 | version-ipfix) template *template-name*]** hierarchy level.

  NOTE: For an MPLS-over-UDP flow that is encapsulated in an RSVP-TE LSP, configure **mpls-ipvx-template** in Junos OS Release 18.1 or **mpls-template** starting in Junos OS 18.2R1 at the **[edit services flow-monitoring (version 9 | version-ipfix) template template-name]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**

Configuring Flow Aggregation to Use IPFIX Flow Templates on PTX Series Routers | **520**

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | **513**

Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers | **465**

ipv6-flow-table-size

Syntax

```
ipv6-flow-table-size units;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services ipv6 flow-table-size]
```

Description

Configure the size of the IPv6 flow table in units of 256K entries.

NOTE: Prior to Junos OS Release 15.1F2, any changes in the configured size of the flow table initiates an automatic reboot of the FPC.

NOTE: The recommended flow table size is 4 so that it can scale up to 4x256K flows, which is 1M. You can configure more, however, the system will issue a warning message.

NOTE: Starting with Junos OS Release 17.3R1, the maximum number of 256K flow entries that you can configure for IPv4 flow tables and IPv6 flow tables is 256 on MPC5Es and MPC6Es with with 4 GB DDR3 memory or higher. The maximum number of 256K flow entries that you can configure for IPv4 flow tables and IPv6 flow tables is 245 on MPC5Es and MPC6Es with DDR3 memory lower than 4 GB.

Options

units—Number of 256K flow entries available for the IPv6 flow table.

Range: 1 through 245

Default: If number of units is not specified, 1024 flow entries are allocated for IPv6.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

ipv6-extended-attrib

Syntax

```
ipv6-extended-attrib;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services ipv6 flow-table-size]
```

Description

Enable the inclusion of element ID, 54, fragmentIdentification, and element ID, 64, ipv6ExtensionHeaders, in IPFIX flow templates that are exported to the flow collector

> **NOTE:** Collection of IPv4 fragmentation IDs occurs automatically without having to configure this setting explicitly.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

ipv6-template

Syntax

```
ipv6-template;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced in Junos OS Release 9.4.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level added in Junos OS Release 10.2.
Statement introduced in Junos OS Release 15.1F4 for PTX Series routers with third-generation FPCs installed. Supported at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.

Description

Specify that the flow aggregation version 9 or IPFIX template is used only for IPv6 records.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**

Configuring Flow Aggregation to Use IPFIX Flow Templates on PTX Series Routers | **520**

jflow-log (Interfaces)

Syntax

```
jflow-log {
    message-rate-limit messages-per-second;
}
```

Hierarchy Level

```
[edit interfaces interface-name services-options]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure generation of log messages or template records in flow monitoring format for NAT error events. These records for NAT error events are generated when addresses for allocation from the NAT pool are not available, when ports for allocation to a subscriber are not available, or when the allocated quota is exceeded for NAT events (more than the configured number of ports is requested).

The remaining statement is described separately.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

jflow-log (Services)

Syntax

```
jflow-log {
    collector collector-name {
        source-ip address;
        destination-address address;
        destination-port port-number;
    }
    collector-group collector-group-name {
        [collector-name1 collector-name2];
    }
    template-profile template-profile-name {
        collector collector-name ;
        collector-group collector-group-name ;
        template-type nat;
        version (ipfix | v9);
        refresh-rate packets packets seconds seconds;
    message-rate-limit messages-per-second
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 14.2R2.

Description

Enable the mechanism to record logging messages in flow monitoring format for NAT events. For this transmission of flow monitoring logs to work properly, the services PIC interface must have an IP address and appropriate logging options configured.

You can configure MX Series routers with MS-MPCs and MS-MICs to log network address translation (NAT) events using the Junos Traffic Vision (previously known as Jflow) version 9 or IPFIX (version 10) template format. This method of generating flow monitoring records for NAT events, such as NAT44 and NAT64 session creation and deletion, and NAT44 and NAT64 binding information base events, enables cohesive and streamlined analysis of NAT traffic and troubleshooting of NAT-related problems.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

label-position

Syntax

```
label-position [ positions ];
```

Hierarchy Level

[edit **services flow-monitoring version9 template** *template-name* **mpls-ipv4-template**],
[edit **services flow-monitoring version9 template** *template-name* **mpls-template**]

Release Information
Statement introduced in Junos OS Release 8.3.

Description
Specify positions for up to three labels in the active flow monitoring version 9 template.

Default
[**1 2 3**]

Options
positions—Numbered positions for the labels.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**

license-server

Syntax

```
license-server {
    ip-address address;
    log-interval seconds;
    services (jflow | cgnat | firewall);
}
```

Hierarchy Level

```
[edit]
```

Release Information

Statement introduced in Junos OS Release 15.1 for MX Series routers.

Description

On MX Series routers with MS-MICs and MS-MPCs, configure the capability to transmit the throughput details per service for the Junos Address Aware, Junos Traffic Vision, and Junos Network Secure services in the last time interval to an external log collector.

Options

ip-address *address*—Use the specified IP address of the license log server.

log-interval *seconds*—Use the specified frequency at which throughput data must be sent from the router to the log collector.

Range: 60 through 86,400 seconds

services—Specify the services for which throughput data must be exported.

- **jflow**—Use inline flow monitoring service or Junos Traffic Vision.
- **cgnat**—Use carrier-grade NAT service or Junos Address Aware.
- **firewall**—Use stateful firewall or Junos Network Secure.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

License Server Management for Throughput Data Export on MX Series Routers for NAT, Firewall, and Inline Flow Monitoring Services | 643

local-dump

Syntax

```
(local-dump | no-local-dump);
```

Hierarchy Level

[edit forwarding-options **sampling instance** *instance-name* **family** (inet |inet6 |mpls) **output flow-server** *hostname*],
[edit forwarding-options **sampling family** (inet |inet6 |mpls) **output flow-server** *hostname*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Enable collection of cflowd records in a log file.

Options

no-local-dump—Do not dump cflowd records to a log file before exporting.

local-dump—Dump cflowd records to a log file before exporting.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

logical-system

Syntax

```
logical-system logical-system-name {
    [ routing-instances instance-name ];
}
```

Hierarchy Level

```
[edit services rpm bgp]
```

Release Information

Statement introduced in Junos OS Release 7.6.

Description

Specify the logical system used by the probes.

Options

logical-system-name—Logical system name.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

match

Syntax

```
match expression;
```

Hierarchy Level

[edit forwarding-options **port-mirroring traceoptions file**],
[edit forwarding-options **sampling traceoptions file**]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the regular expression for lines to be logged for tracing.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | **547**

Configuring Traffic Sampling on MX, M and T Series Routers | **375**

max-connection-duration

Syntax

```
max-connection-duration hours;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information
Statement introduced in Junos OS Release 11.1.

Description
Specify the maximum time a connection can exist between a client and the server.

Options
hours—Number of hours a connection can exist between a client and the server.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

max-duplicates

Syntax

```
max-duplicates number;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify the maximum number of content destinations to which the DFC PIC can send data from a single input set of packets. Limiting the number of duplicates reduces the load on the PIC. This setting overrides the globally applied **g-max-duplicates** setting.

Default

If no value is configured, a default setting of 3 is used.

Options

number—Maximum number of content destinations.

Range: 1 through 64

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

g-max-duplicates | 982

Limiting the Number of Duplicates of a Packet | 328

max-packets-per-second

Syntax

```
max-packets-per-second number;
```

Hierarchy Level

```
[edit forwarding-options sampling input],
[edit forwarding-options sampling instance instance-name input]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the traffic threshold that must be exceeded before packets are dropped. A value of **0** instructs the Packet Forwarding Engine not to sample any traffic.

> **NOTE:** The **max-packets-per-second** statement is not supported when you configure inline flow monitoring (by including the **inline-jflow** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6) output]** hierarchy level).

> **NOTE:** When you configure active monitoring and specify a Monitoring Services, Adaptive Services, or Multiservices PIC in the **output** statement, the **max-packets-per-second** value is ignored.

Options

number—Maximum number of packets per second.

Range: 0 through 65,535

Default: 1000

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

maximum-age

Syntax

```
maximum-age minutes;
```

Hierarchy Level

```
[edit services flow-collector transfer-log-archive]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Maximum age of transfer log file.

Options

minutes—Transfer log file age.

Range: 1 through 360

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

maximum-connections

Syntax

```
maximum-connections count;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the maximum number of allowed connections between the server and all control client hosts.

> **NOTE:** The maximum number of connections between the server and all control client hosts must be greater than or equal to the number of connections between the server and a single controlled client host.

Options

count—Maximum number of connections.

Range: 1 through 1000

Default: 64

Required Privilege Level

system—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | **610**

maximum-connections-per-client

Syntax

```
maximum-connections-per-client count;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the maximum number of allowed connections between the server and a single control client host.

> **NOTE:** The maximum number of connections between the server and all control client hosts must be greater than or equal to the number of connections between the server and a single controlled client host.

Options

count—Maximum number of connections.

Range: 1 through 500

Default: 64

Required Privilege Level

system—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

maximum-packet-length

Syntax

```
maximum-packet-length bytes;
```

Hierarchy Level

```
[edit forwarding-options analyzer analyzer-name input],
[edit forwarding-options port-mirroring input],
[edit forwarding-options port-mirroring instance instance-name input],
[edit forwarding-options sampling input],
[edit forwarding-options sampling instance instance-name input]
```

Release Information

Statement introduced in Junos OS Release 9.6.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport routers. For PTX Series routers with third-generation FPCs installed, **maximum-packet-length** is not supported at the **[edit forwarding-options sampling input]** and **[edit forwarding-options sampling instance *instance-name* input]** hierarchy levels.
For MX Series routers except the MX 80, support at the **[edit forwarding-options analyzer analyzer-name input]** hierarchy level was introduced in Junos OS Release 14.1
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Set the maximum packet length to be used for port mirroring or traffic sampling. Packets longer than the maximum are truncated. This statement cannot be used with inline flow monitoring (**[edit forwarding-options sampling instance *instance-name* family (inet | inet6) output inline-jflow]** .

> **NOTE:** For MX Series routers with Modular Port Interface Concentrators (MPCs), when **maximum-packet-length** (clip length) is configured for port-mirrored packets and the mirror-destination interface is a next-hop-group, the clip length is effective only for the first member interface of the next-hop-group. The mirrored packet copy sent to the rest of the interfaces is not clipped.
>
> In addition, native analyzer sessions (that is, the **[edit forwarding-options analyzer analyzer-name input]** hierarchy level for MX Series routers) can be configured without specifying input parameters. As such, these instances use the following input values by default: rate = 1, and maximum-packet-length = 0.

Options

bytes—Maximum length (in bytes) of the mirrored packet or the sampled packet.

Set the maximum-packet-length value to zero to disable truncation; that is, to mirror or sample the entire packet. Otherwise, Juniper recommends that you configure the packet length to be equal to, or greater than, the IP header length. For IPv4, set the maximum length to at least 20, and for IPv6, set the maximum length to at least 40.

Range: 0 through 9216. For MX Series routers with MPCs, and for EX9200 switches, the range is 1 through 255 bytes.

Default: 0

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring

Configuring Traffic Sampling on MX, M and T Series Routers | 375

maximum-sessions

Syntax

```
maximum-sessions count;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the maximum number of allowed test sessions the server can have running at one time.

> **NOTE:** The maximum number of test sessions running on the server at one time must be greater than or equal to the maximum number of sessions the server can open on a single client connection.

Options

count—Maximum number of sessions.

Range: 1 through 2048

Default: 64

Required Privilege Level

system—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

maximum-sessions-per-connection

Syntax

```
maximum-sessions-per-connection count;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the maximum number of allowed sessions the server can open on a single client connection.

> **NOTE:** The maximum number of test sessions running on the server at one time must be greater than or equal to the maximum number of sessions the server can open on a single client connection.

Options

count—Maximum number of sessions.

Default: 64

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | **610**

media-loss-rate

Syntax

```
media-loss-rate {
    no-syslog-generation;
    generate-snmp-traps;
    storm-control {
        count number;
        interval number;
    }
    alarm-mode {
        immediate;
    }
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure the media loss rate. The media loss rate is the number of media packets lost over a configurable time interval (interval-duration) where the flow packets are packets carrying streaming application information. A single IP packet can contain zero or more streaming packets.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

alarms | 869

media-rate-variation

Syntax

```
media-rate-variation {
    no-syslog-generation;
    generate-snmp-traps;
    storm-control {
        count number;
        interval number;
    }
    alarm-mode {
        mdi-records-count number;
        average;
    }
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure the media rate variation. The media rate variation is the difference between the expected packet rate and actual packet rate expressed as a percentage of the expected packet rate.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

alarms | 869

message-rate-limit (Flow Monitoring Logs for NAT)

Syntax

```
message-rate-limit messages-per-second
```

Hierarchy Level

```
[edit interfaces interface-name services-options jflow-log]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Define the maximum number of logs or template records in flow monitoring format to be generated for NAT error events per second from the specified interface. These records for NAT error events are generated when addresses for allocation from the NAT pool are not available, when ports for allocation to a subscriber are not available, or when the allocated quota is exceeded for NAT events (more than the configured number of ports is requested).

NOTE: The **message-rate-limit** option can be configured only for multiservices interfaces (ms-*x*/*x*/*x*) and not with other interface types.

Options

messages-per-second——Maximum number of flow monitoring log messages per second for NAT error events that can be formatted and sent from the PIC to an external collector. The default rate is 10,000 for an external collector.

Range: 1 through 2,147,483,647

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**

Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**

Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | 307

Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | 309

minimum-priority

Syntax

```
minimum-priority value;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name control-source identifier]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify the minimum priority for the control source.

Options

value—Minimum priority value; if not specified, defaults to 0.

Range: 0 through 254

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

mode (RFC 2544 Benchmarking)

Syntax

```
mode reflect;
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the test mode for the packets that are sent during the benchmarking test.

Options

reflect—Reflect the test frames on the chosen service (IPv4 or Ethernet).

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

monitoring

Syntax

```
monitoring name {
    family inet {
        output {
            cflowd hostname port-number;
            export-format cflowd-version-5;
            flow-active-timeout seconds;
            flow-export-destination {
                (cflowd-collector | collector-pic);
            }
            flow-inactive-timeout seconds;
            interface interface-name {
                number;
                engine-type number;
                input-interface-index number;
                output-interface-index number;
                source-address address;
            }
        }
    }
}
```

Hierarchy Level

```
[edit forwarding-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the flow monitoring instance name and properties.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

moving-average-size

Syntax

```
moving-average-size number;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced in Junos OS Release 8.5.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement Introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Statement at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Enable statistical calculation operations to be performed across a configurable number of the most recent samples.

Options

number—Number of samples to be used in calculations.

Range: 0 through 255

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

mpls-flow-table-size

Syntax

```
mpls-flow-table-size units;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services flow-table-size]
```

Release Information

Statement introduced in Junos OS Release 16.1 for MX Series routers.

Description

Configure the size of the MPLS flow table in units of 256,000 entries.

> **NOTE:** Starting with Junos OS Release 17.3R1, the maximum number of 256K flow entries that you can configure for IPv4 flow tables and IPv6 flow tables is 256 on MPC5Es and MPC6Es with with 4 GB DDR3 memory or higher. The maximum number of 256K flow entries that you can configure for IPv4 flow tables and IPv6 flow tables is 245 on MPC5Es and MPC6Es with DDR3 memory lower than 4 GB.

> **NOTE:** The recommended flow table size is 4 so that it can scale up to 4x256K flows, which is 1M. You can configure more, however, the system will issue a warning message.

Options

units—Number of 256,000 flow entries available for the MPLS flow table.

Range: 1 through 245

Default: 15 (3,840,000)

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

mpls-ipv4-template

Syntax

```
mpls-ipv4-template {
    label-position [ positions ];
}
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced in Junos OS Release 8.3.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level introduced in Junos OS Release 16.1.

Description

Specify the flow aggregation version 9 or IPFIX properties for templates that combine IPv4 and MPLS records. The remaining statement is explained separately.

Starting in Junos OS Release 18.4R1, use **mpls-template** instead of **mpls-ipv4-template** for inline flow monitoring of MPLS-IPv4 flows on the MX Series. You must also configure **tunnel-observation ipv4** at the **[edit services flow-monitoring (version-ipfix | version9) template *template-name*]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers | 465

Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers | 465

mpls-ipvx-template

Syntax

```
mpls-ipvx-template;
```

Hierarchy Level

[edit services flow-monitoring version9 template *template-name*],
[edit services flow-monitoring version-ipfix template *template-name*]

Release Information

Statement introduced in Junos OS Release 18.1R1 on PTX Series routers.

Description

In Junos OS Release 18.1, specify the version 9 or IPFIX template for inline monitoring an MPLS-over-UDP flow that is encapsulated in an RSVP-TE LSP on PTX Series routers. This monitoring looks past the tunnel header to report the inner payload of the packets. To use the template for MPLS-over-UDP flows, you must also configure **tunnel-observation mpls-over-udp** at the **[edit services flow-monitoring (version 9 | version-ipfix) template *template-name*]** hierarchy level.

Starting in Junos OS Release 18.2R1, use **mpls-template** instead of **mpls-ipvx-template**.

> **NOTE:** For an MPLS-over-UDP flow that is carried between IPv4 endpoints, configure **ipv4-template** at the **[edit services flow-monitoring (version9 | version-ipfix) template template-name]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers | 465

mpls-template

Syntax

```
mpls-template {
    label-position [positions ];
}
```

Hierarchy Level

[edit **services flow-monitoring version9 template** *template-name*],
[edit **services flow-monitoring version-ipfix template** *template-name*]

Release Information

Statement introduced in Junos OS Release 8.3.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level introduced in Junos OS Release 16.1.
Statement introduced in Junos OS Release 18.2R1 on PTX Series routers.

Description

Specify the flow aggregation IPFIX or version 9 properties for templates used only for MPLS records.

Starting in Junos OS Release 18.2R1, you can also use **mpls-template** to specify the version 9 or IPFIX template for inline monitoring an MPLS-over-UDP flow that is encapsulated in an RSVP-TE LSP on PTX Series routers. (In Junos OS Release 18.1, use **mpls-ipvx-template** instead of **mpls-template**.) This monitoring looks past the tunnel header to report the inner payload of the packets. To use the template for MPLS-over-UDP flows, you must also configure **tunnel-observation mpls-over-udp** at the **[edit services flow-monitoring (version 9 | version-ipfix) template *template-name*]** hierarchy level.

> **NOTE:** For an MPLS-over-UDP flow that is carried between IPv4 endpoints, configure **ipv4-template** at the **[edit services flow-monitoring (version9 | version-ipfix) template template-name]** hierarchy level.

Starting in Junos OS Release 18.4R1, use **mpls-template** instead of **mpls-ipv4-template** for inline flow monitoring of MPLS-IPv4 flows on the MX Series. You must also configure **tunnel-observation ipv4** at the **[edit services flow-monitoring (version-ipfix | version9) template *template-name*]** hierarchy level.

The remaining statement is explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

multiservice-options

Syntax

```
multiservice-options {
    (core-dump | no-core-dump);
    (syslog | no-syslog);
    flow-control-options {
        down-on-flow-control;
        dump-on-flow-control;
        reset-on-flow-control;
    }
}
```

Hierarchy Level

```
[edit interfaces mo-fpc/pic/port]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For flow-monitoring interfaces only, configure multiservice-specific interface properties.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

name-format

Syntax

```
name-format “format”;
```

Hierarchy Level

```
[edit services flow-collector file-specification variant variant-number]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Specify the name format for a specific file format. The files can include supported macros. Use macros to organize files on the external machine to which they are exported from the collector PIC.

Options
format—Specify the filename format, within quotation marks. The name format can include the following macros, typed in braces:

- **{%D}**—Date
- **{%T)**—Time when the file is created
- **{%I}**—Description string for the logical interface configured using the **collector** statement at the **[edit services flow-collector interface-map]** hierarchy level
- **{%N}**—Unique, sequential number for each new file created
- **{am_pm}**—**AM** or **PM**
- **{date}**—Current date using the**{year} {month} {day}** macros
- **{day}**—From**01** through **31**
- **{day_abbr}**—**Sun** through **Sat**
- **{day_full}**—**Sunday** through **Saturday**
- **{generation number}**—Unique, sequential number for each new file created
- **{hour_12}**—From **01** through **12**
- **{hour_24}**—From **00** through **23**
- **{ifalias}**—Description string for the logical interface configured using the **collector** statement at the **[edit services flow-collector interface-map]** hierarchy level
- **{minute}**—From **00** through **59**

- **{month}**—From **01**through **12**
- **{month_abbr}**—**Jan** through **Dec**
- **{month_full}**—**January** through **December**
- **{num_zone}**—From **-2359** through **+2359**; this macro is not supported
- **{second}**—From **00** through **60**
- **{time}**—Time the file is created, using the **{hour_24} {minute} {second}** macros
- **{time_zone}**—Time zone code name of the locale; for example,**gmt** (this macro is not supported).
- **{year}**—In the format YYYY; for example, **1970**
- **{year_abbr}**—From **00** through **99**

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring File Formats | 272

next-hop (Forwarding Options)

Syntax

```
next-hop address;
```

Hierarchy Level

```
[edit forwarding-options port-mirroring family (inet | inet6) output interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the next-hop address for sending copies of packets to an analyzer.

Options

address—IP address of the next-hop router.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

next-hop-group (Forwarding Options)

Syntax

```
next-hop-group group-name {
   interface interface-name {
      next-hop address;
   }
}
```

Hierarchy Level

```
[edit forwarding-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the next-hop address for sending copies of packets to an analyzer.

It is implicitly assumed that a subgroup is up only if more than one interface in the subgroup is up.

> **NOTE:** In Junos OS releases earlier through Release 14.2, the **next-hop-group** statement is present in the **forwarding-options** stanza for a routing instance, but the **next-hop-group** statement is not allowed in a routing instance. In other words, in a routing instance, **[edit routing-instances *routing-instance-name* forwarding-options next-hop-group]** is not supported. You will get an error message if you try to commit this type of configuration. Starting in Junos OS Release 14.2, the **next-hop-group** statement is not present in **[edit routing-instances *routing-instance-name* forwarding-options]**.

Options

address—IP address of the next-hop router. Each next-hop group supports up to 16 next-hop addresses. Up to 30 next-hop groups are supported. Each next-hop group must have at least two next-hop addresses.

group-name—Name of next-hop group. Up to 30 next-hop groups are supported for the router. Each next-hop group is expected to have at least two next-hop addresses.

interface-name—Name of interface used to reach the next-hop destination.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

next-hop-group (Port Mirroring)

Syntax

```
next-hop-group group-name;
```

Hierarchy Level

```
[edit forwarding-options port-mirroring family (inet | vpls) output],
[edit forwarding-options port-mirroring instance instance-name family (inet | vpls) output]
```

Release Information
Statement introduced in Junos OS Release 9.6.

Description
Specify the next-hop address for sending copies of packets to an analyzer. This configuration enables multipacket port mirroring on MX Series routers and EX Series switches without the use of a Tunnel PIC.

The commit operation fails when a next-hop group has only one interface configured. It is implicitly assumed that a subgroup is up only if more than one interface in the subgroup is up.

Options
group-name—Name of next-hop group.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Port Mirroring with Next-Hop Groups | 552

nexthop-learning

Syntax

```
nexthop-learning (disable | enable);
```

Hierarchy Level

```
[edit services flow-monitoring(version-ipfix |version9) template template-name]
```

Release Information

Statement introduced in Junos OS Release 15.1F2.
Statement introduced in Junos OS Release 17.4R1 for PTX Series routers.
nexthop-learning is supported for **bridge-template** in Junos OS Release 18.2R1 for MX Series routers.

Description

Enable learning of next-hop addresses to correctly report the next hop address, output SNMP, destination IP address, and destination IP mask values in the flow records when a destination is reachable through multiple paths. By default, this behavior of learning the next-hop addresses is disabled for inline flow monitoring. When learning next-hop addresses is disabled, data is reported as follows:

- If the destination address of the sampled IPv4 flow is reachable through multiple paths, the ipNextHopIPv4Address (Element ID 15) and egressInterface (Element ID 14) in the IPv4 flow record are set to the gateway IP address and SNMP index of the first path seen in the forwarding table.
- If the destination address of the sampled IPv6 flow is reachable through multiple paths, the ipNextHopIPv6Address (Element ID 62) and egressInterface (Element ID 14) in the IPv6 flow records are set to 0.
- The Incoming Interface (IIF) and Outgoing Interface (OIF) should be part of the same VRF. If the OIF is in a different VRF, destinationIPv4PrefixLength (Element ID 13), bgpDestinationAsNumber (Element ID 17), ipNextHopIPv4Address (Element ID 15), and egressInterface (Element ID 14) are set to 0 in IPv4 flow records and destinationIPv6PrefixLength (Element ID 30), bgpDestinationAsNumber (Element ID 17), ipNextHopIPv6Address (Element ID 62), and egressInterface (Element ID 14) are set to 0 in IPv6 flow records.

When learning of next-hop addresses is enabled, output SNMP, destination IP address, and destination IP mask values in the flow records are reported correctly. In addition, when enabled, mplsTopLabelIPv4Address (Element ID 47) in IPv4 flow records reports correctly when MPLS ingress sampling is enabled.

To enable next-hop learning, include the **nexthop-learning enable** statement at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.

NOTE: Nexthop learning is only supported when sampling is implemented on the PFE. This is known as inline active flow monitoring. Nexthop learning will not work when sampling is implemented on the MS-DPC/MS-MPC/MS-MIC service cards.

Options

disable—Disable the learning of next hop information required for inline jflow.

enable—Enable the learning of next hop information required for inline jflow.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Next-Hop Address Learning on MX Series Routers for Destinations Accessible Over Multiple Paths | 544

no-filter-check

Syntax

```
no-filter-check;
```

Hierarchy Level

```
[edit forwarding-options port-mirroring family (inet | inet6) output]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Disable filter checking on the port-mirroring interface.

This statement is required when you send port-mirrored traffic to a Tunnel PIC that has a filter applied to it.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

no-remote-trace (Trace Options)

Syntax

```
no-remote-trace;
```

Hierarchy Level

[edit forwarding-options **port-mirroring traceoptions**],
[edit forwarding-options **sampling traceoptions**]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Disable remote tracing.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Tracing Traffic Sampling Operations | 383

no-syslog

Syntax

```
no-syslog;
```

Hierarchy Level

[edit services dynamic-flow-capture **capture-group** *client-name* **control-source** *identifier*]

Release Information
Statement introduced in Junos OS Release 7.4.

Description
Disable system logging of control protocol requests and responses. By default, these messages are logged.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring System Logging | 326

no-syslog-generation

Syntax

```
no-syslog-generation;
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Disable system log generation.

> **NOTE:** If this statement is not configured, **edit services** generates a system log with respective severity level for values not within the configured range.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827
alarms | 869

notification-targets

Syntax

```
notification-targets address port port-number;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name control-source identifier]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

List the destination IP addresses and User Datagram Protocol (UDP) ports to which DFC PICs log exception information and control protocol state transitions, such as timeout values.

Options

address—Allowed destination IP address.

port *port-number*—Allowed destination UDP port number.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

observation-domain-id

Syntax

observation-domain-id *domain-id*;

Hierarchy Level

[edit **services flow-monitoring version-ipfix template** *template-name*]

Release Information

Statement introduced in Junos OS Release 14.1.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

For IPFIX flows, an identifier of an Observation Domain is locally unique to an exporting process of the templates. The export process uses the Observation Domain ID to uniquely identify to the collection process in which the flows were metered. We recommend that you configure this ID to be unique for each IPFIX flow. A value of 0 indicates that no specific Observation Domain is identified by this information element. Typically, this attribute is used to limit the scope of other information elements. If the observation domain is not unique, the collector cannot uniquely identify an IPFIX device.

If you configure the same Observation Domain ID for different template types, such as for IPv4 and IPv6, it does not impact flow monitoring because the actual or the base observation domain ID is transmitted in the flow. The actual observation domain ID is derived from the value you configure and also in conjunction with other parameters such as the slot number, lookup chip (LU) instance, Packet Forwarding Engine instance. Such a method of computation of the observation domain ID ensures that this ID is not the same for two IPFIX devices.

Options

domain-id—Unique identifier for the observation domain for IPFIX flows.

Range: 0 through 255

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows | 527

offload-type

Syntax

```
offload-type {
    none;
    pfe-timestamp;
}
```

Hierarchy Level

[edit services rpm probe *owner* test *test-name*]

Release Information

Statement introduced in Junos OS Evolved Release 20.1

Description

Enable timestamping of RPM probe messages in the Packet Forwarding Engine host processor. This feature is supported only with **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, and **udp-ping-timestamp** probe types.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

one-way-hardware-timestamp

Syntax

```
one-way-hardware-timestamp;
```

Hierarchy Level

[edit **services** rpm **probe** *owner* **test** *test-name*]

Release Information

Statement introduced in Junos OS Release 8.5.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 19.1 for PTX Series routers.
Statement introduced in Junos OS Release 19.2R1 for MPC10E-15C-MRATE on MX240, MX480, and MX960 routers.
Statement introduced in Junos OS Release 19.2R1 for MPC11E on MX2008, MX2010, and MX2020 routers.

Description

Enable timestamping of RPM probe messages for one-way delay and jitter measurements. You must configure this statement along with the **destination-interface** statement to invoke timestamping. This feature is supported only with **icmp-ping**, **icmp-ping-timestamp**, **udp-ping**, and **udp-ping-timestamp** probe types.

NOTE: Starting in Junos OS Evolved Release 20.1R1, the function provided by this command has been replaced by the **offload-type (none|pfe-timestamp)** command.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches | **602**
destination-interface | **916**
hardware-timestamp | **986**

option-refresh-rate

Syntax

```
option-refresh-rate packets packets seconds seconds;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name],
```

Release Information

Statement introduced in Junos OS Release 8.3.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level added in Junos OS Release 10.2.
Statement introduced in Junos OS Release 15.1F4 for PTX Series routers with third-generation FPCs installed. Supported at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.
Support at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level added in Junos OS Release 16.1 for MPLS traffic flows.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Specify the refresh rate, in either packets or seconds.

Options

packets—Refresh rate, in number of packets.

Range: 1 through 480,000

Default: 4800

seconds—Refresh rate, in number of seconds.

Range: 10 through 600

Default: 600

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

options-template-id

Syntax

```
options-template-id id;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced in Junos OS Release 14.1.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Define a unique options template ID to be used for flow aggregation of version 9 and IPFIX flows. If you do not configure values for the template ID and options template ID, default values are assumed for these IDs, which are different for the various address families. If you configure the same template ID or options template ID value for different address families, such a setting is not processed properly and might cause unexpected behavior. For example, if you configure the same template ID value for both IPv4 and IPv6, the collector validates the export data based on the template ID value that it last receives. In this case, if IPv6 is configured after IPv4, the value is effective for IPv6 and the default value is used for IPv4.

Options

id—Unique identifier for the options template to be used for version 9 or IPFIX flows.

Range: 1024 through 65,535

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows | 527
Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows | 531

output (Accounting)

Syntax

```
output {
    aggregate-export-interval seconds;
    cflowd hostname {
        aggregation {
            autonomous-system;
            destination-prefix;
            protocol-port;
            source-destination-prefix {
                caida-compliant;
            }
            source-prefix;
        }
        autonomous-system-type (origin | peer);
        (local-dump | no-local-dump);
        port port-number;
        source-address address;
        version format;
    }
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    interface interface-name {
        engine-id number;
        engine-type number;
        source-address address;
    }
}
```

Hierarchy Level

```
[edit forwarding-options accounting name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure cflowd, output interfaces, and flow properties.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Discard Accounting | 390

output (Monitoring)

Syntax

```
output {
    cflowd hostname port port-number;
    export-format format;
    flow-active-timeout seconds;
    flow-export-destination {
        (cflowd-collector | collector-pic);
    }
    flow-inactive-timeout seconds;
    interface interface-name {
        engine-id number;
        engine-type number;
        input-interface-index number;
        output-interface-index number;
        source-address address;
    }
}
```

Hierarchy Level

```
[edit forwarding-options monitoring name family inet]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure cflowd, output interfaces, and flow properties.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

output (Port Mirroring)

Syntax

```
output {
   interface interface-name {
      next-hop address;
   }
   no-filter-check;
}
```

Hierarchy Level

```
[edit forwarding-options port-mirroring family (inet | inet6)]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure output interfaces and flow properties.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

output (Sampling)

Syntax

```
output {
    aggregate-export-interval seconds;
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    extension-service service-name;
    flow-server hostname {
        aggregation {
            autonomous-system;
            destination-prefix;
            protocol-port;
            source-destination-prefix {
                caida-compliant;
            }
            source-prefix;
        }
        autonomous-system-type (origin | peer);
        dscp dscp-value;
        forwarding-class class-name;
        (local-dump | no-local-dump);
        port port-number;
        source-address address;
        version format;
        version9 {
            template template-name;
        }
    }
    interface interface-name {
        engine-id number;
        engine-type number;
        source-address address;
    }
    file {
        disable;
        filename filename;
        files number;
        size bytes;
        (stamp | no-stamp);
        (world-readable | no-world-readable);
    }
    inline-jflow {
        source-address address;
```

```
        flow-export-rate rate;
    }
}
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls)],
[edit forwarding-options sampling family (inet |inet6 |mpls | vpls)]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Configure cflowd or flow monitoring, output files and interfaces, and flow properties.

The remaining statements are explained separately. See CLI Explorer.

> **NOTE:** The **inline-jflow** statement is valid only under the **[edit forwarding-options sampling instance *instance-name* family inet output]** hierarchy level. The **file** statement is valid only under the **[edit forwarding-options sampling family inet output]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

output-interface-index

Syntax

```
output-interface-index number;
```

Hierarchy Level

```
[edit forwarding-options monitoring name output interface interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify a value for the output interface index that overrides the default supplied by SNMP.

Options

number—Output interface index value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

packet-size (RFC 2544 Benchmarking)

Syntax

```
packet-size bytes;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarking profiles test-profile profile-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.

Description

Define up to 10 packet sizes that are used sequentially for the test.

Options

bytes—Size of the test packet. If you enter multiple packet sizes, you must separate each number with a space.

Range: 64 through 9136 bytes

NOTE: The valid packet sizes are 64, 68, 72, 128, 256, 512, 768, 1024, 1280, 1518, 1522, 1600, 1728, 2496, 3584, 4016, 9104, and 9136 bytes. If you specify a packet size other than the ones listed here as valid sizes, the configuration is saved when you commit the setting and no error message is displayed. However, when you start the test by entering the **test services rpm rfc2544-benchmarking test *test-name* start** command, an error message is displayed if you configured an invalid packet size in the test profile associated with the test name.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

RFC 2544-Based Benchmarking Tests Overview | **763**

Configuring RFC 2544-Based Benchmarking Tests | **770**

rfc2544-benchmarking | **1108**

passive-monitor-mode

Syntax

```
passive-monitor-mode;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For Asynchronous Transfer Mode (ATM), SONET/SDH, Fast Ethernet, and Gigabit Ethernet interfaces only, monitor packet flows from another router. If you include this statement in the configuration, the SONET/SDH interface does not send keepalives or alarms, and does not participate actively on the network.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Passive Flow Monitoring on M Series, MX Series or T Series Routers | 203

multiservice-options | 1048

password (Flow Collector File Servers)

Syntax

```
password "password";
```

Hierarchy Level

```
[edit services flow-collector destination ftp:url]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the primary and secondary destination FTP server password.

Options

password—FTP server password.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Destination FTP Servers for Flow Records | 271

password (Transfer Log File Servers)

Syntax

```
password "password";
```

Hierarchy Level

```
[edit services flow-collector transfer-log-archive archive-sites]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the primary and secondary destination FTP server password.

Options

password—FTP server password.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

peer-as-billing-template

Syntax

```
peer-as-billing-template;
```

Hierarchy Level

[edit **services flow-monitoring version9 template** *template-name*]

Release Information
Statement introduced in Junos OS Release 10.4.

Description
Enables the extraction of bandwidth usage information for billing purposes in PIC-based sampling configurations. This capability is supported on routers and applies only to IPv4 and IPv6 traffic.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**

persistent-results

Syntax

```
persistent-results;
```

Hierarchy Level

```
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced in Junos OS Release 19.1.

Description

When enabled, allows to display the old and current tests' results after a network recovery or after TWAMP server reachability when you execute the **show services rpm twamp client probe-results** command. By default, the option is disabled.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

Understanding Two-Way Active Measurement Protocol on Routers | 582

pic-memory-threshold

Syntax

```
pic-memory-threshold percentage percentage;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Specify a PIC memory usage percentage that triggers a system log warning message.

Options

percentage—PIC memory threshold value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Thresholds | 328

pop-all-labels

Syntax

```
pop-all-labels {
    required-depth number;
}
```

Hierarchy Level

```
[edit interfaces interface-name atm-options mpls],
[edit interfaces interface-name fastether-options mpls],
[edit interfaces interface-name gigether-options mpls],
[edit interfaces interface-name sonet-options mpls]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For passive monitoring on ATM, SONET/SDH, Fast Ethernet, and Gigabit Ethernet interfaces only, removes up to two MPLS labels from incoming IP packets. For passive monitoring on T Series devices, removes up to five MPLS labels from incoming IP packets. For passive monitoring on MX Series routers with MPCs, all labels are popped by default and the **required-depth** statement is ignored.

Except for MX Series routers with MPCs, this statement has no effect on IP packets with more than two MPLS labels, or IP packets with more than five MPLS labels on T Series devices. Packets with MPLS labels cannot be processed by the monitoring PIC; if packets with MPLS labels are forwarded to the monitoring PIC, they are discarded.

The remaining statement is explained separately. See CLI Explorer.

Default

If you omit this statement, the MPLS labels are not removed, and the packet is not processed by the monitoring PIC.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Passive Flow Monitoring for MPLS Encapsulated Packets | 205

port (Flow Monitoring)

Syntax

```
port port-number;
```

Hierarchy Level

```
[edit forwarding-options accounting name output cflowd hostname],
[edit forwarding-options monitoring name family inet output cflowd hostname],
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls) output flow-server hostname],
[edit forwarding-options sampling family (inet | inet6 | mpls) output flow-server hostname]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the User Datagram Protocol (UDP) port number on the cflowd host system or flow server.

Options

port-number—Any valid UDP port number on the host system.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Flow Aggregation | 489

port (RPM)

Syntax

```
port number;
```

Hierarchy Level

```
[edit services rpm probe-server (tcp | udp)]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.

Description

Specify the port number for the probe server.

Options

number—Port number for the probe server. The value can be **7** or **49,160** through **65,535**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Receiver Servers | **601**

port (TWAMP)

Syntax

```
port number;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

TWAMP server listening port.

Options

number—Port number.

Range: 1 through 65,535

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

port-mirroring

Syntax

```
port-mirroring {
    input {
        maximum-packet-lengthbytes
        rate rate;
        run-length number;
    }
    family any {
        output {
            (next-hop-group group-name | interface interface-name);
        }
    }
    family inet {
        output {
            interface interface-name {
                next-hop address;
            }
            no-filter-check;
        }
    }
    instance instance-name {
        input {
            rate rate;
            maximum-packet-length number;
        }
        family any {
            output {
                (next-hop-group group-name | interface interface-name);
            }
        }
        family inet {
            output {
                next-hop-group group-name;
            }
        }
    }
    traceoptions {
        file filename <files number> <size bytes> <world-readable | no-world-readable>;
    }
}
```

Hierarchy Level

[edit forwarding-options]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the input, output, and traceoptions properties for sending copies of packets to an analyzer.

NOTE: Option **run-length** is not supported on MX Series routers with MPCs.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

post-cli-implicit-firewall

Syntax

```
post-cli-implicit-firewall;
```

Hierarchy Level

```
[edit services rpm twamp]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Ensure that the CLI configured (**explicit firewall**) takes precedence over the implicit firewall. The inline TWAMP client or server uses implicit firewall to achieve its functionality.

> **NOTE:** Wrong configuration of CLI firewall can lead to improper functioning of inline TWAMP client or server. After you enable or disable this configuration statement, you must restart the router, or restart remote operation using the command **restart remote-operations**, for the operation to be effective.
>
> On issuing the command **restart remote-operations** all TWAMP sessions (both client and server) are aborted. You must restart all the RPM sessions and all TWAMP sessions (both client and server).

Default

The default for this configuration statement is in disabled status.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Two-Way Active Measurement Protocol on Routers | 582

pre-rewrite-tos

Syntax

```
pre-rewrite-tos;
```

Hierarchy Level

```
[edit forwarding-options sampling]
```

Release Information
Statement introduced in Junos OS Release 14.1.

Description
Preserve prenormalized type-of-service (ToS) value for egress sampled or mirrored packets. This configuration preserves the prerewrite ToS value for all forms of sampling, such as Routing Engine-based sampling, port mirroring, flow monitoring, and so on. This statement is effective for egress sampling only.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

probe

Syntax

```
probe owner {
    test test-name {
        data-fill data;
        data-size size;
        delegate-probes probes;
        destination-interface interface-name;
        destination-port port;
        dscp-code-point dscp-bits;
        history-size size;
        inet6-options source-address ipv6-address;
        moving-average-size number;
        next-hop next-hop;
        offload-type {
            none;
            pfe-timestamp;
        }
        probe-count count;
        probe-interval seconds;
        probe-type type;
        routing-instance instance-name;
        rpm-scale {
            destination {
                interface interface-name.logical-unit-number;
                subunit-cnt subunit-cnt;
            }
            source {
                address-base ipv4-address-base;
                count ipv4-count;
                step ipv4-step;
            }
            source-inet6 {
                address-base ipv6-address-base;
                count ipv6-count;
                step ipv6-step;
            }
            target {
                address-base ipv4-address-base;
                count ipv4-count;
                step ipv4-step;
            }
            target-inet6 {
```

```
            address-base ipv6-address-base;
            count ipv6-count;
            step ipv6-step;
          }
          tests-count tests-count;
        }
        source-address address;
        target (url url | address ipv4-address | inet6-url url | inet6-address ipv6-address);
        test-interval interval;
        thresholds
        {
          egress-time microseconds;
          ingress-time microseconds;
          jitter-egress microseconds;
          jitter-ingress microseconds;
          jitter-rtt microseconds;
          rtt microseconds;
          std-dev-egress microseconds;
          std-dev-ingress microseconds;
          std-dev-rtt microseconds;
          successive-loss count;
          total-loss count;
        }
        traps [trap-names];
        ttl [hop-count];
    }
}
```

Hierarchy Level

```
[edit services rpm]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify an owner name. The owner name combined with the test name represent a single RPM configuration instance.

Options

owner—Owner name up to 32 characters in length.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

probe-count

Syntax

```
probe-count count;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name test-session session-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the number of probes within a test.

Options

count—1 through 15 for RPM, for TWAMP 1 through 4,294,967,290.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | **629**

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

probe-interval

Syntax

```
probe-interval interval;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name test-session session-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify the time to wait between sending packets, in seconds.

Options

interval—Number of seconds, from 1 through 255.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring BGP Neighbor Discovery Through RPM | **629**
- Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**
- Understanding Two-Way Active Measurement Protocol on Routers | **582**

probe-limit

Syntax

```
probe-limit limit;
```

Hierarchy Level

```
[edit services rpm]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Configure the maximum number of concurrent probes allowed.

Options

limit—Maximum number of concurrent probes allowed.

Range: (MX Series routers only) Starting in Junos OS Release 17.2R2 and 17.3R1, 1 through 2000. In Junos releases earlier than 17.2R1, the range is 1 through 500.

Range: (PTX Series Packet Transport routers only) 1 through 200

Range: (Other platforms) 1 through 500

Default: 100

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Limiting the Number of Concurrent RPM Probes on M, MX, T and PTX Routers and EX Series Switches | 602

probe-server

Syntax

```
probe-server {
   tcp {
      destination-interface interface-name;
      port number;
   }
   udp {
      destination-interface interface-name;
      port number;
   }
}
```

Hierarchy Level

```
[edit services rpm]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify the server to act as a receiver for the probes.

The remaining statements are explained separately. See CLI Explorer.

> **NOTE:** The **destination-interface** statement is not supported on PTX Series routers.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Receiver Servers | 601

probe-type

Syntax

```
probe-type type;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.

Description

Specify the packet and protocol contents of a probe.

Options

type—One of the following probe type values:

- **http-get**—(Not available at the **[edit services rpm bgp]** hierarchy level.) Sends a Hypertext Transfer Protocol (HTTP) get request to a target URL.
- **http-metadata-get**—(Not available at the **[edit services rpm bgp]** hierarchy level.) Sends an HTTP get request for metadata to a target URL.
- **icmp-ping**—Sends ICMP echo requests to a target address.
- **icmp-ping-timestamp**—Sends ICMP timestamp requests to a target address.
- **tcp-ping**—Sends TCP packets to a target.
- **udp-ping**—Sends UDP packets to a target.
- **udp-ping-timestamp**—Sends UDP timestamp requests to a target address.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

rate

Syntax

```
rate new-sessions-per-second;
```

Hierarchy Level

```
[edit interfaces interface-name services-options session-limit]
```

Release Information

Statement introduced in Junos OS Release 9.6.
Support added in Junos OS Release 19.3R2 for Next Gen Services on MX Series routers MX240, MX480 and MX960 with the MX-SPC3 services card.

Description

Specify the maximum number of new sessions allowed per second on services cards.

Options

rate *new-sessions-per-second*—Specify the maximum number of new sessions allowed per second.

Range: 0, which indicates no limit, or greater.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

profiles (RFC 2544 Benchmarking)

Syntax

```
profiles {
   test-profile profile-name {
      test-type (throughput | latency | frame-loss | back-back-frames);
      packet-size bytes;
      step-percent percent;
      bandwidth-kbps kpbs;
   }
}
```

Hierarchy Level

[edit services rpm rfc2544-benchmarking]

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.

Description

Configure the test profile to specify attributes, such as the period for the test and the type of test to be performed, for the RFC 2544-based benchmarking test. The test profile is referenced in the test interface to perform a specific type of benchmarking test and compute statistics to describe the performance characteristics of a network interconnecting device.

Options

profiles—Define the test profile for the RFC 2544-based benchmarking test to examine and analyze the performance characteristics of a network interconnecting device.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- RFC 2544-Based Benchmarking Tests Overview | 763
- Configuring RFC 2544-Based Benchmarking Tests | 770
- rfc2544-benchmarking | 1108

rate (Forwarding Options)

Syntax

```
rate number;
```

Hierarchy Level

```
[edit forwarding-options analyzer analyzer-name input],
[edit forwarding-options port-mirroring family (inet | inet6) input],
[edit forwarding-options port-mirroring input],
[edit forwarding-options sampling input],
[edit forwarding-options sampling instance instance-name input]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport routers.
Support at the **[edit forwarding-options analyzer analyzer-name input]** hierarchy level for MX Series routers introduced in Junos OS Release 14.1.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Set the ratio of the number of packets to be sampled. For example, if you specify a rate of 10, every tenth packet (1 packet out of 10) is sampled.

Native analyzer sessions (that is, the **[edit forwarding-options analyzer analyzer-name input]** hierarchy level for MX Series routers) can be configured without specifying input parameters, which means that the instance uses default input values: rate = 1 and maximum-packet-length = 0.

> **NOTE:** The recommended sampling rate for the MX150 is 1000 or greater. If you configure less than 1000, a warning is issued.

Options

number—Denominator of the ratio.

Range: 1 through 16000000(16M)

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring

Configuring Traffic Sampling

receive-options-packets

Syntax

```
receive-options-packets;
```

Hierarchy Level

[edit **interfaces** *interface-name* **unit** *logical-unit-number* **family** inet]

Release Information
Statement introduced before Junos OS Release 7.4.

Description
When you enable passive monitoring, this statement is required for conformity with cflowd records structure.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Passive Flow Monitoring on M Series, MX Series or T Series Routers | 203

receive-ttl-exceeded

Syntax

```
receive-ttl-exceeded;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number family inet]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

When you enable passive monitoring, this statement is required for conformity with cflowd records structure.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Enabling Passive Flow Monitoring on M Series, MX Series or T Series Routers | 203

refresh-rate (Flow Monitoring Logs for NAT)

Syntax

```
refresh-rate packets packets seconds seconds;
```

Hierarchy Level

```
[edit services jflow-log template-profile template–profile-name]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the refresh rate for transmitting flow template records with version 9 and IPFIX templates for NAT events to the collector, in either packets or seconds.

Options

packets— Number of packets after which templates are sent to the collector.

Range: 1 through 480,000

Default: 4800

seconds—Number of seconds after which templates are sent to the collector

Range: 10 through 600

Default: 600

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

reflect-mode (RFC2544 Benchmarking)

Syntax

```
reflect-mode (mac-rewrite | mac-swap | no-mac-swap );
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Specify the reflection mode for the benchmarking test.

Options

mac-rewrite—(ACX Series routers only) Enable rewriting of the MAC address on the reflected frames. The MAC addresses specified in the source-mac-address and destination-mac-address options are used.

mac-swap—Swap the source and destination MAC addresses in the test frame. This is the default behavior.

NOTE: In bridge families, when the service type is **ELAN**, MAC addresses are swapped by default, on the reflected frames. And, when the service type is **ELINE** , MAC addresses are not swapped by default.

no-mac-swap—Do not swap the source and destination MAC addresses in the test frame. The frame is returned to the originator without any modification to the MAC addresses.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests for E-LAN and E-Line Services on MX Series Routers | **652**

rfc2544-benchmarking | **1108**

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647
Configuring an RFC 2544-Based Benchmarking Test | 659

reflect-etype (RFC 2544 Benchmarking)

Syntax

```
reflect-etype ethertype-value;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 15.1 for MX104 Universal Routing Platforms.

Description

Specify the EtherType to be used for reflection of the test frames. EtherType is a two-octet field in an Ethernet frame that defines the protocol in the frame payload. This statement is valid only if you configure the test mode to be a reflector. If you do not configure this statement, all EtherTypes are reflected.

Options

ethertype-value—Identifier for the EtherType. The EtherType value appears in the Ethernet type field of the packet. It specifies the protocol being transported in the Ethernet frame. For instance, the EtherType for IPv4 is 0x0800. So, if you specify the value as 2048, IPv4 packets are reflected.

Range: 1 through 65,535

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647
Supported RFC2544-Based Benchmarking Statements on MX Series Routers | 657
Configuring an RFC 2544-Based Benchmarking Test | 659

required-depth

Syntax

```
required-depth number;
```

Hierarchy Level

```
[edit interfaces interface-name atm-options mpls pop-all-labels],
[edit interfaces interface-name fastether-options mpls pop-all-labels],
[edit interfaces interface-name gigether-options mpls pop-all-labels],
[edit interfaces interface-name sonet-options mpls pop-all-labels]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

For passive monitoring on ATM, SONET/SDH, Fast Ethernet, and Gigabit Ethernet interfaces only, specify the number of MPLS labels an incoming packet must have for the **pop-all-labels** statement to take effect. For passive monitoring on MX Series routers with MPCs, all labels are popped by default and the **required-depth** statement is ignored.

If you include the **required-depth 1** statement, the **pop-all-labels** statement takes effect for incoming packets with one label only. If you include the **required-depth 2** statement, the **pop-all-labels** statement takes effect for incoming packets with two labels only.

Options

number—Number of MPLS labels on incoming IP packets.

Range: 1 through 2 labels.

Default: If you omit this statement, the **pop-all-labels** statement takes effect for incoming packets with one or two labels. The default is equivalent to including the **required-depth [1 2]** statement.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Passive Flow Monitoring for MPLS Encapsulated Packets | 205

Junos OS Network Interfaces Library for Routing Devices

retry (Services Flow Collector)

Syntax

```
retry number;
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the maximum number of attempts the flow collector interface make to transfer log files to the FTP server.

Options

number—Maximum number of transfer retry attempts.

Range: 0 through 10

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Retry Attempts | 274

retry-delay

Syntax

```
retry-delay seconds;
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the amount of time the flow collector interface waits between retry attempts.

Options

seconds—Amount of time between transfer retry attempts.

Range: 0 through 60

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Retry Attempts | 274

rfc2544-benchmarking

Syntax

```
rfc2544-benchmarking {
   profiles {
      test-profile profile-name {
         test-type (throughput | latency | frame-loss | back-back-frames);
         packet-size bytes;
         step-percent percent;
         bandwidth-kbps kpbs;
      }
   }
   tests{
      test-name test-name {
         test-interface interface-name;
         mode reflect;
         family (bridge| inet | ccc);
         destination-ipv4-address address;
         destination-udp-port port-number;
         source-ipv4-address address;
         source-udp-port port-number;
         direction (egress | ingress);
      }
   }
}
```

Hierarchy Level

```
[edit services rpm]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Configure the parameters for the RFC 2544-based benchmarking test. You must configure a test profile, which specifies the type of test and the manner in which it must be performed, and associate the test profile with a test name. The test name that you configure contains details, such as the address family and the test mode, for the test. You can associate the same test profile with multiple test names.

Define the attributes for the RFC 2544-based benchmarking test to examine and analyze the performance characteristics of a network interconnecting device.

The remaining statements are explained separately.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- show services rpm rfc2544-benchmarking | **1371**
- show services rpm rfc2544-benchmarking test-id | **1377**

rfc6514-compliant-safi129 (Protocols BGP)

Syntax

```
rfc6514-compliant-safi129
```

Hierarchy Level

```
[edit logical-systems logical-system-name protocols bgp],
[edit logical-systems logical-system-name protocols bgp group group-name],
[edit logical-systems logical-system-name protocols bgp group group-name neighbor neighbor-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp group group-name],
[edit logical-systems logical-system-name routing-instances routing-instance-name protocols bgp group group-name
    neighbor neighbor-name],
[edit protocols bgp],
[edit protocol bgp group group-name],
[edit protocols bgp group group-name neighbor neighbor-name],
[edit routing-instances routing-instance-name protocols bgp],
[edit routing-instances routing-instance-name protocols bgp group group-name],
[edit routing-instances routing-instance-name protocols bgp group group-name neighbor neighbor-name]
```

Release Information

Statement introduced in Junos OS Release 16.1 for MX Series routers.

Description

Parse and send BGP VPN multicast traffic according to Subsequent Address Family Identifier (SAFI) 129, as defined in RFC 6514 (that is, *length*, *prefix*). The Network Layer Reachability Information (NLRI) format used for BGP VPN multicast in previous releases of Junos OS was SAFI 128, which was *length*, *label*, *prefix*.

Required Privilege Level

routing—To view this statement in the configuration.
routing-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

routing-instance

Syntax

```
routing-instance instance-name;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name]
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the routing instance used by the probes. The routing instance is also applicable for control connection.

NOTE: The media interface from where the TWAMP control and test or data packets arrive and exit the **si- logical interface** must be a part of the same routing instance.

Options

instance-name—Routing instance configured at the **[edit routing-instance]** hierarchy level.

Default: Internet (IPv4) routing table **inet.0**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591
Understanding Two-Way Active Measurement Protocol on Routers | 582

routing-instance (cflowd)

Syntax

```
routing-instance instance-name;
```

Hierarchy Level

```
[edit forwarding-options sampling family (inet | inet6 | mpls) output flow-server hostname]
```

Release Information

Statement introduced in Junos OS Release 13.3.

Description

Configure a non-default VPN routing and forwarding (VRF) instance through which flow collectors can be reachable for inline flow monitoring. You cannot configure a flow collector to be reachable through non-default VRF instances for version 5 and version 8 flows. You must configure the routing instance to be a VRF instance by including the **instance-type vrf** statement at the **[edit routing-instances instance-name]** hierarchy level.

Options

instance-name—Name of a routing instance that has been configured at the **[edit routing-instance]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Directing Traffic Sampling Output to a Server Running the cflowd Application

routing-instance-list (TWAMP)

Syntax

```
routing-instance-list {
    instance-name {
        port number;
    }
}
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure the Two-Way Active Measurement Protocol (TWAMP) servers on specific routing instances, instead of associating the TWAMP server at the system-level to apply to all routing instances configured on a router. The default routing instance is Internet routing table **inet.0**. If you do not specify a routing instance, the TWAMP probe applies to all routing instances. To apply the TWAMP probe to only the default routing instance, you must explicitly set the value of instance-name to default. If an interface is not part of any routing instance, the default port is used for TWAMP probes. You can configure up to 100 routing instances for a TWAMP server.

Options

instance-name—Name of the routing instance, a maximum of 31 characters.

number—Port number.

Range: 1 through 65,535

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

routing-instances

Syntax

```
routing-instances instance-name;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm bgp logical-system logical-system-name]
```

Release Information

Statement introduced in Junos OS Release 7.6.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.

Description

Specify the routing instance used by the probes.

Options

instance-name—A routing instance configured at the **[edit routing-instances]** hierarchy level.

Default: Internet routing table **inet.0**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

rpm (Interfaces)

Syntax

```
rpm (client client | server server | twamp-client twamp-client | twamp-server twamp-server);
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number]
```

Release Information

Statement introduced in Junos OS Release 8.1.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 15.1 for MX Series routers.

Description

Associate an RPM or TWAMP client (router or switch that originates RPM or TWAMP probes) or RPM or TWAMP server with a specified interface.

NOTE: The TWAMP client is applicable only for **si-** interfaces.

Options

client—Identifier for RPM client router or switch.

server—Identifier for RPM server.

twamp-client—Identifier for RPM TWAMP client router.

twamp-server—Identifier for RPM TWAMP server.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches | 602

rpm (Services)

Syntax

```
rpm {
    bgp {
        data-fill data;
        data-size size;
        destination-port port;
        history-size size;
        logical-system logical-system-name [routing-instances routing-instance-name];
        moving-average-size number;
        probe-count count;
        probe-interval seconds;
        probe-type type;
        routing-instances instance-name;
        test-interval interval;
    }
    probe owner {
        test test-name {
            data-fill data;
            data-size size;
            destination-interface interface-name;
            destination-port port;
            dscp-code-point dscp-bits;
            hardware-timestamp;
            history-size size;
            moving-average-size number;
            one-way-hardware-timestamp;
            probe-count count;
            probe-interval seconds;
            probe-type type;
            routing-instance instance-name;
            source-address address;
            target (url url | address address);
            test-interval interval;
            thresholds thresholds;
            traps traps;
        }
    }
    probe-server {
        tcp {
            destination-interface interface-name;
            port number;
        }
```

```
    udp {
        destination-interface interface-name;
        port number;
    }
}
probe-limit limit;
rfc2544-benchmarking {
    profiles {
        test-profile profile-name {
            test-type (throughput | latency | frame-loss | back-back-frames);
            packet-size bytes;
            step-percent percent;
            bandwidth-kbps kpbs;
        }
    }
    tests{
        test-name test-name {
            test-interface interface-name;
            mode reflect;
            family (bridge| inet | ccc);
            destination-ipv4-address address;
            destination-udp-port port-number;
            source-ipv4-address address;
            source-udp-port port-number;
            direction (egress | ingress);
        }
    }
}
traceoptions {
    file filename <files number> <match regular-expression > <size maximum-file-size> <world-readable |
        no-world-readable>;
    flag flag;
}
```

```
    twamp {
        server {
            authentication-mode (authenticated | encrypted | none);
            authentication-key-chain identifier {
                key-id identifier {
                    secret password-string;
                }
            }
            client-list list-name {
                [ address address ];
            }
            inactivity-timeout seconds;
            maximum-connections-duration hours;
            maximum-connections count;
            maximum-connections-per-client count;
            maximum-sessions count;
            maximum-sessions-per-connection count;
            port number;
            routing-instance-list {
                instance-name {
                    port number;
                }
            }
            server-inactivity-timeout minutes;
            tcp-keepcnt
            tcp-keepidle
            tcp-keepintvl
        }
    }
    rfc2544-benchmarking {
        tests{
            test-name test-name {
                test-interface interface-name;
                mode reflect;
                family (inet | ccc);
                destination-ipv4-address address;
                destination-udp-port port-number;
                source-ipv4-address address;
                source-udp-port port-number;
                direction (egress | ingress);
            }
        }
    }
}
```

Hierarchy Level

[edit services]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure BGP neighbor discovery through RPM.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring BGP Neighbor Discovery Through RPM | 629

rpm-scale

Syntax

```
rpm-scale {
    destination {
        interface interface-name.logical-unit-number;
        subunit-cnt subunit-cnt;
    }
    source {
        address-base ipv4-address-base;
        count ipv4-count;
        step ipv4-step;
    }
    source-inet6 {
        address-base ipv6-address-base;
        count ipv6-count;
        step ipv6-step;
    }
    target {
        address-base ipv4-address-base;
        count ipv4-count;
        step ipv4-step;
    }
    target-inet6 {
        address-base ipv6-address-base;
        count ipv6-count;
        step ipv6-step;
    }
    tests-count tests-count;
}
```

Hierarchy Level

[edit **services** rpm **probe** *owner* **test** *test-name*]

Release Information

Statement introduced in Junos OS Release 17.4R1 on MX Series routers.

Description

Configure the generation of multiple IPv4 RPM tests for a probe owner. Starting in Junos OS Release 18.2R1, you can also configure the generation of multiple IPv6 RPM tests for a probe owner. Tests are generated for multiple combinations of source and target addresses, which are incremented based on your

configuration. Tests are first generated for all the source addresses with the initial target address, then tests are generated for all the source addresses with the next available target address, and so on.

Options

interface-name.logical-unit-number—The services interface that is generating RPM probes and the logical unit number that is used for the first test that is generated.

ipv4-address-base—The IPv4 source or target address that is incremented to generate the addresses used in the RPM tests.

ipv4-count—The maximum number of IPv4 source or target addresses to use for the generated RPM tests.

ipv4-step—The amount to increment the IPv4 source or target address for each generated RPM test.

ipv6-address-base—The IPv6 source or target address that is incremented to generate the addresses used in the RPM tests.

ipv6-count—The maximum number of IPv6 source or target addresses to use for the generated RPM tests.

ipv6-step—The amount to increment the IPv6 source or target address for each generated RPM test.

subunit-cnt—The maximum number of logical units used by the services interface in the generated tests. The first generated test uses the logical unit specified in the *interface-name.logical-unit-number* option, and each successive test increments the logical unit number by one. Once the maximum number of logical units has been used, the next generated test cycles back to the logical unit that was used in the first test.

tests-count—The maximum number of RPM tests to generate. This number must be less than or equal to the number of generated source addresses multiplied by the number of generated target addresses.

Range: 1 through 500,000 for probes generated on an MS-MPC or MS-MIC. 1 through 2,000 for probes generated on the Routing Engine.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

run-length

Syntax

```
run-length number;
```

Hierarchy Level

```
[edit forwarding-options port-mirroring input],
[edit forwarding-options port-mirroring instance port-mirroring-instance-name input],
[edit forwarding-options port-mirroring family (inet|inet6) input],
[edit forwarding-options sampling input],
[edit forwarding-options sampling instance instance-name input]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport routers.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Set the number of samples following the initial trigger event. The configuration enables you to sample packets following those already being sampled.

> **NOTE:** The **run-length** statement is not supported when you configure inline flow monitoring (by including the **inline-jflow** statement at the **[edit forwarding-options sampling instance *instance-name* family (inet | inet6) output]** hierarchy level).

Options

number—Number of samples.

Range: 0 through 20

Default: 0

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Applying Forwarding Table Filters

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

Configuring Traffic Sampling on MX, M and T Series Routers | 375

sample-once

Syntax

```
sample-once;
```

Hierarchy Level

[edit forwarding-options **sampling**]

Release Information

Statement introduced in Junos OS Release 9.6.

Description

Explicitly sample a packet for active monitoring only once. Setting this option avoids duplication of packets in cases where sampling is enabled at both the ingress and egress interfaces and simplifies analysis of the sampled traffic.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

sampling (Forwarding Options)

Syntax

```
sampling {
    disable;
    family (inet | inet6 | mpls | vpls) {
        disable;
        output {
            aggregate-export-interval seconds;
            extension-service service-name;
            file {
                disable;
                filename filename;
                files number;
                size bytes;
                (stamp | no-stamp);
                (world-readable | no-world-readable);
            }
            flow-active-timeout seconds;
            flow-inactive-timeout seconds;
            flow-server hostname {
                aggregation {
                    autonomous-system;
                    destination-prefix;
                    protocol-port;
                    source-destination-prefix {
                        caida-compliant;
                    }
                    source-prefix;
                }
                autonomous-system-type (origin | peer);
                (local-dump | no-local-dump);
                port port-number;
                source-address address;
                version format;
                version9 {
                    template template-name;
                }
            }
            interface interface-name {
                engine-id number;
                engine-type number;
                source-address address;
            }
```

```
    }
}
input {
    max-packets-per-second number;
    maximum-packet-length bytes;
    rate number;
    run-length number;
}
```

```
instance instance-name {
    disable;
    family (bridge | inet | inet6 | mpls | vpls) {
        disable;
        output {
            aggregate-export-interval seconds;
            extension-service service-name;
            flow-server hostname {
                aggregation {
                    autonomous-system;
                    destination-prefix;
                    protocol-port;
                    source-destination-prefix {
                        caida-compliant;
                    }
                    source-prefix;
                }
                autonomous-system-type (origin | peer);
                dscp dscp-value;
                forwarding-class class-name;
                (local-dump | no-local-dump);
                port port-number;
                source-address address;
                version format;
                version9 {
                    template template-name;
                }
                version-ipfix {
                    template template-name;
                }
            }
            inline-jflow {
                source-address address;
                flow-export-rate rate;
            }
            interface interface-name {
                engine-id number;
                engine-type number;
                source-address address;
            }
        }
    }
    input {
        max-packets-per-second number;
```

```
        maximum-packet-length bytes;
        rate number;
        run-length number;
      }
    }
    pre-rewrite-tos;
    sample-once;
    traceoptions {
      no-remote-trace;
      file filename <files number> <size bytes> <match expression> <world-readable | no-world-readable>;
    }
}
```

Hierarchy Level

```
[edit forwarding-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.
Statement introduced in Junos OS Release 16.1X65 for PTX1000 routers.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Configure traffic sampling.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

Applying Forwarding Table Filters

Collecting Traffic Sampling Output in the Cisco Systems NetFlow Services Export Version 9 Format

Directing Traffic Sampling Output to a Server Running the cflowd Application

Configuring Port Mirroring

Tracing Traffic-Sampling Operations

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

sampling (Interfaces)

Syntax

```
sampling direction;
```

Hierarchy Level

```
[edit interfaces interface-name unit logical-unit-number family inet],
[edit logical-systems logical-system-name interfaces interface-name unit logical-unit-number family inet]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the direction of traffic to be sampled.

Options

direction can be one of the following:

input—Configure at least one expected ingress point.

output—Configure at least one expected egress point.

input output—On a single interface, configure at least one expected ingress point and one expect egress point.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Junos OS Services Interfaces Library for Routing Devices

Configuring Flow Monitoring | 3

sampling-instance

Syntax

```
sampling-instance instance-name;
```

Hierarchy Level

```
[edit chassis fpc slot-number],
[edit chassis lcc number fpc slot-number] (Routing Matrix),
[edit chassis member member-number fpc slot slot-number]
```

Release Information

Statement introduced in Junos OS Release 9.6.
Support at the **[edit chassis member *member-number* fpc slot *slot-number*]** hierarchy level introduced in Junos OS Release 14.1.
Statement introduced in Junos OS Release 14.1R3 for EX Series switches.

Description

Associate a defined sampling instance with a specific FPC, MPC, or DPC for active sampling instances configured at the **[edit forwarding-options sampling]** hierarchy level.

For M120 routers with FEB, this statement must also be configured under **[edit chassis feb *slot-number*]**, in addition to the **[edit forwarding-options sampling]** hierarchy level.

In a two-member MX Series Virtual Chassis, the master router (member 0) uses FPC slot numbers 0 through 11 with no offset; the backup router (member 1) uses FPC slot numbers 12 through 23, with an offset of 12.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Associating Sampling Instances for Active Flow Monitoring with a Specific FPC, MPC, or DPC | **118**

Inline Flow Monitoring for Virtual Chassis Overview

server

Syntax

```
server {
    client-list list-name {
        [ address address ];
    }
    inactivity-timeout seconds;
    maximum-connections count;
    maximum-connections-per-client count;
    maximum-sessions count;
    maximum-sessions-per-connection count;
    port number;
}
```

Hierarchy Level

[edit services rpm **twamp**]

Release Information

Statement introduced in Junos OS Release 9.3.

Description

TWAMP server configuration settings.

Options

The remaining statements are described separately.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | **610**

server-inactivity-timeout

Syntax

```
server-inactivity-timeout minutes;
```

Hierarchy Level

```
[edit services rpm twamp server]
```

Release Information

Statement introduced in Junos OS Release 11.1.

Description

The maximum time the Two-Way Active Measurement Protocol (TWAMP) server has to finish the TWAMP control protocol negotiation.

Options

minutes—Number of minutes the TWAMP server has to finish the TWAMP control protocol negotiation.

Default: 15 minutes

Range: 1 through 30 minutes

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

service-port

Syntax

```
service-port port-number;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name control-source identifier]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Identify the User Datagram Protocol (UDP) port number for control protocol requests.

Options

port-number—Port number for control protocol request messages.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

service-type (RFC2544 Benchmarking)

Syntax

```
service-type (elan | eline) ;
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Mention the service under test. Possible values are **elan** and **eline**. This statement is applicable only for the bridge family or when the **mode** is configured as reflect. When the service type is **elan**, MAC addresses are swapped by default on the reflected frames. The **no-mac-swap** is not supported in this service type. When the service type is **eline**, MAC addresses are not swapped by default in the reflected frames. Use the **mac-swap** option to swap the addresses.

NOTE: When you configure the Layer 2 reflection, you can specify the service type under test as ELINE if you want to simulate an ELINE service using bridge encapsulation.

Options

elan—Specify elan service type.

eline—Specify eline service type.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

rfc2544-benchmarking | **1108**

Configuring an RFC 2544-Based Benchmarking Test | **659**

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

services

Syntax

```
services { ... }
```

Hierarchy Level

```
[edit]
```

Release Information
Statement introduced before Junos OS Release 7.4.

Description
Configure router services.

The underlying statements are explained separately.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

services

Syntax

```
services rpm { ... }
```

Hierarchy Level

```
[edit]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.

Description

Define the service rules to be applied to traffic.

Options

rpm—Use the RPM set of rules statements.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring BGP Neighbor Discovery Through RPM | **629**
- Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**
- Configuring RPM Receiver Servers | **601**
- Limiting the Number of Concurrent RPM Probes on M, MX, T and PTX Routers and EX Series Switches | **602**
- Configuring RPM Timestamping on MX, M, T, and PTX Series Routers and EX Series Switches | **602**
- Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | **610**
- Enabling RPM on MX, M and T Series Routers and SRX Firewalls for the Services SDK | **641**

services-options

Syntax

```
services-options {
    cgn-pic;
    close-timeout
    fragment-limit
    disable-global-timeout-override;
    ignore-errors <alg> <tcp>;
    inactivity-non-tcp-timeout seconds;
    inactivity-tcp-timeout seconds;
    inactivity-timeout seconds
    open-timeout seconds;
    pba-interim-logging-interval seconds;
    reassembly-timeout
    session-limit {
        maximum number;
        rate new-sessions-per-second;
        cpu-load-threshold percentage;
    }
    session-timeout seconds;
    jflow-log {
        message-rate-limit messages-per-second;
    }
    syslog {
        host hostname {
            facility-override facility-name;
            log-prefix prefix-value;
            port port-number;
            services severity-level;
        }
        message-rate-limit messages-per-second;
    }
    tcp-tickles tcp-tickles;
    trio-flow-offload minimum-bytes minimum-bytes;
}
```

Hierarchy Level

```
[edit interfaces interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Define the service options to be applied on an interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

> *Configuring Default Timeout Settings for Services Interfaces.*
>
> *Configuring System Logging for Services Interfaces*

shared-key

Syntax

```
shared-key value;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name control-source identifier]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Configure the authentication key value.

Options

value—Secret authentication value shared between a control source and destination.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

size

Syntax

```
size bytes;
```

Hierarchy Level

```
[edit forwarding-options port-mirroring traceoptions file],
[edit forwarding-options sampling family (inet |inet6 |mpls) output file],
[edit forwarding-options sampling traceoptions file]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the maximum size of each file containing sample or log data. The file size is limited by the number of files to be created and the available hard disk space.

When a traffic sampling file named **sampling-file** reaches the maximum size, it is renamed **sampling-file.0**. When the **sampling-file** again reaches its maximum size, **sampling-file.0** is renamed **sampling-file.1** and **sampling-file** is renamed **sampling-file.0**. This renaming scheme continues until the maximum number of traffic sampling files is reached. Then the oldest traffic sampling file is overwritten.

Options

bytes—Maximum size of each traffic sampling file or trace log file, in kilobytes (KB), megabytes (MB), or gigabytes (GB).

Syntax: ***x*k** to specify KB, ***x*m** to specify MB, or ***x*g** to specify GB

Range: 10 KB through the maximum file size supported on your router

Default: 1 MB for sampling data; 128 KB for log information

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

Configuring Traffic Sampling on MX, M and T Series Routers | 375

slamon-services

Syntax

```
slamon-services rfc2544;
```

Hierarchy Level

```
[edit chassis fpc slot-number]
```

Release Information

Statement introduced in Junos OS Release 16.1R1.

Description

(MX240, MX480, MX960, MX2010, and MX2020 routers only) Enable support for RFC2544-based benchmarking tests on MX Series routers with MPC1 (MX-MPC1-3D), MPC2 (MX-MPC2-3D), and the 16-port 10-Gigabit Ethernet MPC (MPC-3D-16XGE-SFP) that are hosting test interfaces. For aggregated interfaces, enable support for RFC2544-based benchmarking tests on all MPCs hosting child links. A system log is generated when you enable support for RFC2544-based benchmarking tests on unsupported MPCs.

NOTE: On MX104 and MX80 Series routers that have a single fixed FPC, this configuration is not required.

Options

rfc2544—RFC2544-based benchmarking tests.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647
Configuring an RFC 2544-Based Benchmarking Test | 659
Enabling Support for RFC2544-Based Benchmarking Tests on MX Series Routers | 666

soft-limit

Syntax

```
soft-limit bandwidth;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name content-destination identifier]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify a bandwidth threshold at which congestion notifications are sent to each control source of the criteria that point to this content destination. If the control source is configured with the **syslog** statement, a log message also be generated.

Options

bandwidth—Soft limit threshold, in bits per second.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Content Destination | 323

soft-limit-clear

Syntax

```
soft-limit-clear bandwidth;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name content-destination identifier]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Specify a bandwidth threshold at which the latch set by the soft-limit threshold is cleared.

Options

bandwidth—Soft-limit clear threshold, in bits per second.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Content Destination | 323

soft-limit | 1142

source-address (Forwarding Options)

Syntax

```
source-address address;
```

Hierarchy Level

```
[edit forwarding-options accounting name outputinterface interface-name],
[edit forwarding-options monitoring namefamilyfamily inet output interface interface-name],
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls | vpls) output interface
    interface-name],
[edit forwarding-options sampling family (inet |inet6 |mpls) output interface interface-name],
[edit forwarding-options sampling instance instance-name family inet output inline-jflow]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the source address for monitored packets.

Options

address—Interface source address.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring Discard Accounting | 390
- Configuring Flow Monitoring | 3
- Configuring Traffic Sampling on MX, M and T Series Routers | 375

source-address (Services)

Syntax

```
source-address address;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.

Description

Specify the source IP address used for probes. If the source IP address is not one of the router's or switch's assigned addresses, the packet use the outgoing interface's address as its source.

The following addresses cannot be used for the source IP address used for probes:

- 0.0.0.0
- 127.0.0.0/8 (loopback)
- 224.0.0.0/4 (multicast)
- 255.255.255.255 (broadcast)

Options

address—Valid IP address.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

source-addresses

Syntax

```
source-addresses [ addresses ];
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name control-source identifier]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

List the IP addresses from which the control source can send control protocol requests to the Juniper Networks router.

Options

address—Allowed IP source address.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Control Source | 324

source-id

Syntax

```
source-id source-id;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name]
```

Release Information

Statement introduced in Junos OS Release 14.1.

Description

For version 9 flows, a 32-bit value that identifies the Exporter Observation Domain is called the source ID. NetFlow collectors use the combination of the source IP address and the source ID field to separate different export streams originating from the same exporter.

Options

source-id—Unique identifier for the source for version 9 flows.

Range: 0 through 255

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows | 527

Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows | 531

source-ip (Flow Monitoring Logs for NAT)

Syntax

```
source-ip address;
```

Hierarchy Level

```
[edit services jflow-log collector collector-name]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the source IPv4 address of the services PIC interface to be used for generation of flow monitoring log messages in flow monitoring template format for NAT events.

Options

address—Valid IPv4 address.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

source-ipv4-address (RFC 2544 Benchmarking)

Syntax

```
source-ipv4-address address;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the source IPv4 address to be used in generated test frames. This parameter is optional for both **ccc** and **inet** families. If you do not configure the source IPv4 address for an **inet** family, the source address of the interface is used to transmit the test frames.

Options

address—Valid IPv4 address.

Default: If you do not configure the source IPv4 address for a **ccc** family, default value of 192.168.1.10 is used.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

source-mac-address (RFC2544 Benchmarking)

Syntax

```
source-mac-address mac-address;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Specify the source MAC address used in generated test frames. This parameter is applicable for a bridge family.

Options

mac-address—Source MAC address. Specify the MAC address as six hexadecimal bytes in one of the following formats: ***nnnn.nnnn.nnnn*** or ***nn:nn:nn:nn:nn:nn***; for example, **0000:5e00:5355** or **00:00:5e:00:53:55**.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- rfc2544-benchmarking | **1108**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- Configuring an RFC 2544-Based Benchmarking Test | **659**

source-udp-port (RFC 2544 Benchmarking)

Syntax

```
source-udp-port port-number;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X52 for ACX Series routers.
Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the UDP port of the source to be used in the UDP header for the generated frames. If you do not specify the UDP port, the default value of 4041 is used.

Options

port-number—Source UDP port number for the test frames

Default: 4041

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

stamp

Syntax

```
(stamp | no-stamp);
```

Hierarchy Level

```
[edit forwarding-options sampling family (inet |inet6 |mpls) output file]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Include a timestamp with each line in the output file.

Options

no-stamp—Do not include timestamps. This is the default.

stamp—Include a timestamp with each line of packet sampling information.

Default: No timestamp is included.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Traffic Sampling on MX, M and T Series Routers | 375

storm-control

Syntax

```
storm-control {
    count number;
    interval number;
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Configure the count and the interval to control the flooding of SNMP traps per flow.

Options

count *number*—Use the specified maximum number of SNMP traps generated in the configured interval.

interval *number*—Use the specified minimum time period, in seconds, between the generation of successive traps.

Default: The default count value is 1.The default interval is 1 second.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827
alarms | 869

syslog

Syntax

```
(syslog | no-syslog);
```

Hierarchy Level

```
[edit interfaces mo-fpc/pic/port multiservice-options]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

System logging is enabled by default. The system log information of the Monitoring Services PIC is passed to the kernel for logging in the **/var/log** directory.

- **syslog**—Enable PIC system logging.
- **no-syslog**—Disable PIC system logging.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Monitoring | 3

target (Services RPM)

Syntax

```
target (url url | address address);
```

Hierarchy Level

```
[edit services rpm probe owner test test-name]
[edit services rpm twamp client control-connection control-client-name test-session session-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify the destination address or URL used for the probes.

Options

url *url*—For HTTP probe types, use the specified fully formed URL that includes **http://** in the URL address. You can also specify an IPv6 address of a host in the URL to denote the destination or server to which the RPM probes must be sent.

> **NOTE:** The *url* is for RPM only.

address *address*—For all probe types other than the HTTP probes, use the specified IPv4 or IPv6 address for the target host.

> **NOTE:** Starting with Junos OS Release 14.2R2, the RPM client router (the router or switch that originates the RPM probes) can send probe packets to the RPM probe server (the device that receives the RPM probes) that contains an IPv6 address.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591
- Understanding Two-Way Active Measurement Protocol on Routers | 582
- *Configuring the Interface for RPM Timestamping for Client/Server on a Switch (CLI Procedure)*

tcp

Syntax

```
tcp {
    destination-interface interface-name;
    port port;
}
```

Hierarchy Level

```
[edit services rpm probe-server]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify the port information for the TCP server.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring RPM Receiver Servers | 601

tcp-keepcnt

Syntax

```
tcp-keepcnt number;
```

Hierarchy Level

```
[edit services rpm twamp (client control-connection control-client-name |server)]
```

Release Information

Statement introduced in Junos OS Release 19.1.

Description

Number of unacknowledged probes to send before considering the connection dead and notifying the application layer. The range is 1 through 50.

Default

The default number of TCP KEEPALIVEs sent is 6.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610
Understanding Two-Way Active Measurement Protocol on Routers | 582

tcp-keepidle

Syntax

```
tcp-keepidle seconds;
```

Hierarchy Level

```
[edit services rpm twamp (client control-connection control-client-name |server)]
```

Release Information

Statement introduced in Junos OS Release 19.1.

Description

Time interval between the last data packet sent and the first keepalive probe. The range is 1 through 600 seconds.

Default

The default value is 120 seconds.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

Understanding Two-Way Active Measurement Protocol on Routers | 582

tcp-keepintvl

Syntax

```
tcp-keepintvl seconds;
```

Hierarchy Level

```
[edit services rpm twamp (client control-connection control-client-name |server)]
```

Release Information

Statement introduced in Junos OS Release 19.1.

Description

Time interval between successive keepalive probes. The range is 1second through 600 seconds.

Default

The default value is 5 seconds.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

Understanding Two-Way Active Measurement Protocol on Routers | 582

template (Flow Monitoring IPFIX Version)

Syntax

```
template template-name {
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    flow-key {
        flow-direction;
        vlan-id;
        output-interface;
    }
    (bridge-template|ipv4-template | ipv6-template | mpls-ipv4-template | mpls-ipvx-template | vpls-template);
    nexthop-learning (enable |disable);
    observation-domain-id
    option-refresh-rate packets packets seconds seconds;
    options-template-id
    template-id
    template-refresh-rate packets packets seconds seconds;
    tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
}
```

Hierarchy Level

```
[edit services flow-monitoring version-ipfix]
```

Release Information

Statement introduced in Junos OS Release 10.2.
Statement introduced in Junos OS Release 12.R3 for EX Series switches.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Specify the IPFIX output template properties to support flow monitoring.

Options

template-name—Name of the IPFIX template.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | 513

template (Flow Monitoring Version 9)

Syntax

```
template template-name {
    flow-active-timeout seconds;
    flow-inactive-timeout seconds;
    flow-key {
        flow-direction;
        vlan-id;
        output-interface;
    }
    (bridge-template|ipv4-template | ipv6-template | mpls-template |vpls-templatelabel-position [ positions ] |
        mpls-ipv4-template label-position [ positions ] | mpls-ipvx-template);
    option-refresh-rate packets packets seconds seconds;
    options-template-id
    peer-as-billing-template;
    source-id
    template-id
    template-refresh-rate packets packets seconds seconds;
    tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
}
```

Hierarchy Level

```
[edit services flow-monitoring version9]
```

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Specify the version 9 output template properties to support flow monitoring.

Options

template-name—Name of the version 9 template.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

template (Forwarding Options)

Syntax

```
template template-name;
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls) output flow-server hostname
    version9],
[edit forwarding-options sampling family (inet |inet6 |mpls) output flow-server hostname version9]
```

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Specify flow monitoring version 9 template to be used for output of sampling records.

Options

template-name—Name of the version 9 template.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

template (Forwarding Options Version IPFIX)

Syntax

```
template;
```

Hierarchy Level

```
[edit forwarding-options sampling instance family (inet | inet6 | mpls | vpls) output flow-server hostname version-ipfix]
```

Release Information

Statement introduced in Junos OS Release 10.2.
Statement introduced in Junos OS Release 12.R3 for EX Series switches.

Description

Specify flow monitoring version IPFIX properties to apply to output sampling records.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | 513

template (Inline Monitoring)

Syntax

```
template name {
    observation-domain-id observation-domain-id;
    option-template-id option-template-id;
    option-template-refresh-rate option-template-refresh-rate;
    template-id template-id;
    template-refresh-rate template-refresh-rate;
}
```

Hierarchy Level

```
[edit services inline-monitoring]
```

Release Information

Statement introduced in Junos OS 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Configure template for inline packet monitoring.

Options

name—Name of the template.

observation-domain-id—Significant one byte of observation domain ID used to uniquely identify exporting process. Other three bytes are system generated and are unique within the chassis.

Default: 0

Range: 0 through 255

option-template-id—Option template ID.

Default: 640

Range: 1024 through 65535

option-template-refresh-rate—Option refresh rate in seconds.

Default: 600 seconds

Range: 10 through 600 seconds

template-id—Template ID.

Default: 384

Range: 1024 through 65535

template-refresh-rate—Refresh rate in seconds.

Default: 600 seconds

Range: 10 through 600 seconds

Required Privilege Level

system

RELATED DOCUMENTATION

Understanding Inline Monitoring Services | 362

template-id

Syntax

```
template-id id;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced in Junos OS Release 14.1.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Define a template ID to be used for flow aggregation of version 9 and IPFIX flows. If you do not configure values for the template ID and options template ID, default values are assumed for these IDs, which are different for the various address families. If you configure the same template ID or options template ID value for different address families, such a setting is not processed properly and might cause unexpected behavior. For example, if you configure the same template ID value for both IPv4 and IPv6, the collector validates the export data based on the template ID value that it last receives. In this case, if IPv6 is configured after IPv4, the value is effective for IPv6 and the default value is used for IPv4.

Options

id—Unique identifier for the template to be used for version 9 or IPFIX flows.

Range: 1024 through 65,535

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows | 527
Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows | 531

template-profile (Flow Monitoring Logs for NAT)

Syntax

```
template-profile template-profile-name;
```

Hierarchy Level

```
[edit services jflow-log],
[edit services service-set service-set-name jflow-log]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the name of the flow template profile to be used for generating flow monitoring format messages for NAT events and for transmitting them to the collector. You can define a template profile for the Jflow service by using this statement at the **[edit services jflow-log]** hierarchy level, and associate the template profile with a service set by using this statement at the **[edit services service-set *service-set-name* jflow-log]** hierarchy level.

Options

template-profile-name—Name of the flow template profile for NAT events. The name can be up to 32 alphanumeric characters in length. Allowed characters are [a-zA-Z0-9_].

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

template-refresh-rate

Syntax

```
template-refresh-rate packets packets seconds seconds;
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name],
[edit services flow-monitoring version-ipfix template template-name],
```

Release Information

Statement introduced in Junos OS Release 8.3.
Support at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level added in Junos OS Release 10.2.
Statement introduced in Junos OS Release 15.1F4 for PTX Series routers with third-generation FPCs installed. Supported at the **[edit services flow-monitoring version-ipfix template *template-name*]** hierarchy level.
Support at the **[edit services flow-monitoring version9 template *template-name*]** hierarchy level added in Junos OS Release 16.1 for MPLS traffic flows.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Specify the frequency at which the flow generator sends updates about template definitions to the flow collector. Specify either the number of packets or the number of seconds.

Options

packets—Refresh rate, in number of packets.

Range: 1 through 480,000

Default: 4800

seconds—Refresh rate, in number of seconds.

Range: 10 through 600

Default: 600

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

template-type (Flow Monitoring Logs for NAT)

Syntax

```
template-type nat;
```

Hierarchy Level

```
[edit services jflow-log template-profile template–profile-name]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the type of service for which flow template profiles, in version or IPFIX format, must be used for generating flow monitoring format messages for NAT events and for transmitting them to the collector. Currently, you can configure only NAT events or services for generation of log messages in flow monitoring format.

Options

nat—Use flow template profiles for generation of flow monitoring logs for NAT events.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

templates

Syntax

```
templates {
    template-name {
        interval-duration interval-duration;
        inactive-timeout inactive-timeout;
        rate {
            (layer3 layer3-packets-per-second | media media-bits-per-second);
        }
        delay-factor {
            ;
            threshold {
                (info | warning | critical) delay-factor-threshold;
            }
        }
        media-loss-rate {
            disable;
            threshold {
                (info | warning | critical) percentage mlr-percentage | packet-count mlr-packet-count);
            }
        }
        media-rate-variation {
            disable;
            threshold {
                (info | warning | critical) mrv-variation;
            }
        }
        media-packets-count-in-layer3 media-packets-count-in-layer3;
        media-packet-size media-packet-size;
    }
}
```

Hierarchy Level

[edit services **video-monitoring**]

Release Information

Statement introduced in Junos OS Release 14.1.

Description

Configure the media delivery index template containing the measurement parameters for video monitoring.

Options

delay-factor—Define delay factor syslog threshold levels.

delay-factor-threshold—Delay factor threshold in milliseconds. When the threshold is exceeded, a syslog message is generated.

Default: 0—Do not generate syslogs.

Range: 0 though 65,535 milliseconds

disable—Disable logging for the threshold.

inactive-timeout—Number of seconds of flow inactivity after which time media delivery index statistics collection for a flow is terminated.

Range: 30 through 300 seconds

info | warning | critical—Level of syslog message generated when a threshold is exceeded.

interval-duration—Number of seconds after which time media delivery index flow monitoring statistics for the interval are reported.

Range: 1 through 50

layer3-packets-per-second—Layer 3 packet rate in IP packets per second.

Range: 0 though 4,294,967,295 pps

media-bits-per-second—Media bit rate for the stream in bits per second.

media-loss-rate—Define media loss rate syslog threshold levels.

media-packets-count-in-layer-3—Number of media packets in an IP packet.

Range: 1 through 32

media-packet-size—Size of media packet in bits.

Default: 188

Range: 1 through 2048

media-rate-variation—Define delay factor syslog threshold levels.

mlr-packet-count—Media loss rate threshold expressed as the number of packets dropped. When the threshold is exceeded, a syslog message is generated.

mlr-percentage—Media loss rate threshold expressed as the percentage of total packets dropped. When the threshold is exceeded, a syslog message is generated.

Range: 0 through 100

mrv-variation—Media rate variation threshold. The variation is the ratio of actual media rate to the configured media rate, expressed as a percentage.

template-name—Name of the template containing media delivery index measurement criteria. The template can be assigned to an interface.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Video Monitoring on MX Series Routers | 833

test

Syntax

```
test test-name {
    data-fill data;
    data-size size;
    destination-interface interface-name;
    destination-port port;
    dscp-code-point dscp-bits;
    hardware-timestamp;
    history-size size;
    moving-average-size number;
    inet6-options;
    one-way-hardware-timestamp;
    probe-count count;
    probe-interval seconds;
    probe-type type;
    routing-instance instance-name;
    rpm-scale {
        destination {
            interface interface-name.logical-unit-number;
            subunit-cnt subunit-cnt;
        }
        source {
            address-base ipv4-address-base;
            count ipv4-count;
            step ipv4-step;
        }
        source-inet6 {
            address-base ipv6-address-base;
            count ipv6-count;
            step ipv6-step;
        }
        target {
            address-base ipv4-address-base;
            count ipv4-count;
            step ipv4-step;
        }
        target-inet6 {
            address-base ipv6-address-base;
            count ipv6-count;
            step ipv6-step;
        }
        tests-count tests-count;
```

```
    }
    source-address address;
    target (url url | address address);
    test-interval interval;
    thresholds thresholds;
    traps traps;
    ttl hop-count
}
```

Hierarchy Level

```
[edit services rpm probe owner]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
inet6-options option added in Junos OS Release 14.1R4 for MX Series routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify the range of probes over which the standard deviation, average, and jitter are calculated. The test name combined with the owner name represent a single RPM configuration instance.

Options

test-name—Test name. The name can be up to 32 characters in length.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | 591

tests (RFC 2544 Benchmarking)

Syntax

```
tests {
    test-name test-name {
        test-interface interface-name;
        mode reflect;
        family (inet | ccc);
        destination-ipv4-address address;
        destination-udp-port port-number;
        source-ipv4-address address;
        source-udp-port port-number;
        direction (egress | ingress);
    }
}
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarking]
```

Release Information

Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the attributes of the test iteration, such as the address family (type of service, IPv4 or Ethernet), the logical interface, test duration, and test packet size, that are used for a benchmarking test to be run. The test name combined with the test profile represent a single real-time performance monitoring (RPM) configuration instance.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | 659
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647
- rfc2544-benchmarking | 1108

test-interface (RFC 2544 Benchmarking)

Syntax

```
test-interface interface-name;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarkingtests test-name test-name]
```

Release Information

Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Specify the logical interface on which the RFC 2544-based benchmarking test is run. If you configure an **inet** family and the test mode to initiate and terminate test frames on the same device, the interface you configure is not effective. Instead, the test is run on the egress logical interface that is determined using route lookup on the specified destination IPv4 address. If you configure an **inet** family and the test mode to reflect the frames back on the sender from the other end, the logical interface is used as the interface to enable the reflection service (reflection is performed on the packets entering the specified interface). If you not configure the logical interface for reflection test mode, a lookup is performed on the source IPv4 address to determine the interface that hosts the address.

Options

interface-name—Name of the logical interface on which the test needs to be run.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

test-interval

Syntax

```
test-interval seconds;
```

Hierarchy Level

```
[edit services rpm bgp],
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify the time to wait between tests, in seconds. A test interval of 0 seconds causes the RPM test to stop after one iteration.

Options

seconds—Number of seconds to wait between tests.

Range: **[edit services rpm bgp]** and **[edit services rpm probe *owner* test *test-name*]** hierarchy levels: 0 through 86,400

Range: **[edit services rpm twamp client control-connection *control-client-name*]** hierarchy level: 1 through 255

Default: 1

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring BGP Neighbor Discovery Through RPM | **629**
- Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**
- Understanding Two-Way Active Measurement Protocol on Routers | **582**

test-name (RFC 2544 Benchmarking)

Syntax

```
test-name test-name {
    test-interface interface-name;
    mode reflect;
    family (inet | ccc);
    destination-ipv4-address address;
    destination-udp-port port-number;
    source-ipv4-address address;
    source-udp-port port-number;
    direction (egress | ingress);
}
```

Hierarchy Level

[edit services rpm rfc2544-benchmarking tests]

Release Information

Statement introduced in Junos OS Release 13.3 for MX104 Universal Routing Platforms.

Description

Define the name of the RFC 2544-based benchmarking test. For each unique test name that you configure, you can specify a test profile, which contains the settings for a test and its type, and also a test interface, which contains the settings for test packets that are sent and received on the selected interface.

Options

test-name—Test name. The name can be up to 32 characters in length.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | 659
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647
- rfc2544-benchmarking | 1108

test-profile (RFC 2544 Benchmarking)

Syntax

```
test-profile profile-name;
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarking tests test-name test-name]
[edit services rpm rfc2544-benchmarking profiles]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.

Description

Specify the name of the test profile to be associated with a particular test name. This parameter is required when the test mode is configured as initiate-and-terminate. This parameter is disregarded when the test mode is configured as reflection. A reflection service does not use the parameters specified in the test profile.

Options

profile-name—Name of the test profile. The name can be up 32 characters in length. The name must start with a letter. Allowed characters are [a-zA-Z0-9_]

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- RFC 2544-Based Benchmarking Tests Overview | 763
- Configuring RFC 2544-Based Benchmarking Tests | 770
- rfc2544-benchmarking | 1108

test-session

Syntax

```
test-session session-name {
    data-fill-with zeros data;
    data-size size;
    dscp-code-point dscp-bits;
    probe-count count;
    probe-interval seconds;
    target (url url | address address);
}
```

Hierarchy Level

[edit services rpm twamp client control connection *session-name*]

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the test session details that includes the session name, the contents of the test packet, the data size, the probe details, and the target destination details.

Options

session-name—Name of the session.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Understanding Two-Way Active Measurement Protocol on Routers | 582

test-type (RFC 2544 Benchmarking)

Syntax

```
test-type (throughput | latency | frame-loss | back-back-frames);
```

Hierarchy Level

```
[edit services rpm rfc2544-benchmarking profiles test-profile profile-name]
```

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.

Description

RFC 2544 defines four main test types. You can configure and perform a test for a certain service, such as IPv4 or Ethernet, and analyze the results of the test to examine the various SLA parameters of the service. The test packets traverse through the same path as the regular service traffic.

Configure the type of RFC 2544-based benchmarking test to be performed. Because of the ability of these tests to measure throughput, bursty frames, frame loss, and latency, this mechanism is also used to diagnose and examine Ethernet-based networks.

Options

throughput—Measure the maximum rate at which none of the offered or transmitted frames are dropped by the device on which the test is performed.

latency—Measure the time interval between the arrival of the last bit of the input frame at the input port and the output of the first bit of the frame on the output port.

fame-loss—Measure the percentage of frames that must have been forwarded by a network device under steady state (constant) load conditions, but were not forwarded due to lack of resources.

back-back-frames—Measure the number of frames that are forwarded by the device on which the test is performed when a burst of frames with minimum inter-frame gaps is sent to that device from another source device.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

RFC 2544-Based Benchmarking Tests Overview | 763

thresholds

Syntax

```
thresholds thresholds;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Packet Series Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Specify thresholds used for the probes. A system log message is generated when the configured threshold is exceeded. Likewise, an SNMP trap (if configured) is generated when a threshold is exceeded.

> **NOTE:** If you configure a value of zero using the ***thresholds*** option for a certain probe parameter, the generation of SNMP traps is disabled for the corresponding probe attribute. For example, if you specify the **set thresholds jitter-egress 0** statement, it denotes that traps are not triggered when the jitter in egress time threshold is met or exceeded.

Options

thresholds—Specify one or more threshold measurements. The following options are supported:

- **egress-time**—Measures maximum source-to-destination time per probe.
- **ingress-time**—Measures maximum destination-to-source time per probe.
- **jitter-egress**—Measures maximum source-to-destination jitter per test.
- **jitter-ingress**—Measures maximum destination-to- source jitter per test.
- **jitter-rtt**—Measures maximum jitter per test, from 0 through 60,000,000 microseconds.
- **rtt**—Measures maximum round-trip time per probe, in microseconds.
- **std-dev-egress**—Measures maximum source-to-destination standard deviation per test.

- **std-dev-ingress**—Measures maximum destination-to-source standard deviation per test.
- **std-dev-rtt**—Measures maximum standard deviation per test, in microseconds.
- **successive-loss**—Measures successive probe loss count, indicating probe failure.
- **total-loss**—Measures total probe loss count indicating test failure, from 0 through 15.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**
Understanding Two-Way Active Measurement Protocol on Routers | **582**

traceoptions (Dynamic Flow Capture)

Syntax

```
traceoptions {
    file filename <files number> <size size> <world-readable | non-world-readable>;
}
```

Hierarchy Level

```
[edit services dynamic-flow-capture]
```

Release Information

Statement introduced in Junos OS Release 9.2.

Description

Enable and define tracing options for dynamic flow capture events.

Options

file *filename*—Use the specified file to receive the output of the tracing operation. Enclose the name within quotation marks. All files are placed in the directory **/var/log**.

files *number*—(Optional) Use the specified maximum number of trace files. When a trace file named *trace-file* reaches its maximum size, it is renamed *trace-file*.0, then *trace-file*.1, and so on, until the maximum number of trace files is reached. Then, the oldest trace file is overwritten. Note that if you specify a maximum number for files, you must also specify a maximum file size with the **size** option.

Range: 2 through 1000 files.

Default: 10 files.

no-world-readable—(Optional) Restrict access to the file.

world-readable—(Optional) Enable free access to the file.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Junos Capture Vision | 321

traceoptions (Forwarding Options)

Syntax

```
traceoptions {
    no-remote-trace;
    file filename <files number> <size bytes> <match expression> <world-readable | no-world-readable>;
}
```

Hierarchy Level

```
[edit forwarding-options port-mirroring],
[edit forwarding-options sampling]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure traffic sampling tracing operations.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Tracing Traffic Sampling Operations | 383

traceoptions (Inline Monitoring)

Syntax

```
traceoptions {
    file <filename> <files files> <match match> <size size> <(world-readable | no-world-readable)>;
    no-remote-trace;
}
```

Hierarchy Level

```
[edit services inline-monitoring]
```

Release Information

Statement introduced in Junos OS 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Configure traceoptions for the inline monitoring process.

Options

file—Trace file information.

file-name—Name of file in which the trace information is available.

files *files*—Maximum number of trace files.

Default: 3

Range: 2 through 1000

match *match*—Regular expression for lines to be logged.

world-readable—Allow any user to read the log file.

no-world-readable—Don't allow any user to read the log file.

size *size*—Specify maximum trace file size.

Range: 10240 through 1073741824

no-remote-trace—Disable remote tracing.

Required Privilege Level

system

RELATED DOCUMENTATION

Understanding Inline Monitoring Services | 362

traceoptions (RPM)

Syntax

```
traceoptions {
    file filename <files number> <match regular-expression > <size maximum-file-size> <world-readable |
        no-world-readable>;
    flag flag;
}
```

Hierarchy Level

```
[edit services rpm]
```

Release Information

Statement introduced in Junos OS Release 13.2.

Description

Define tracing operations for RPM processes.

Options

file *filename*—Use the specified file to receive the output of the tracing operation. All files are placed in the directory **/var/log**.

Default: rmopd

files *number*—(Optional) Use the specified maximum number of trace files to create before overwriting the oldest one. If you specify a maximum number of files, you also must specify a maximum file size with the **size** option.

Range: 2 through 1000

Default: 3 files

match *regular-expression*—(Optional) Use the specified regular expression to refine the output to include lines that contain the regular expression.

size *maximum-file-size*—(Optional) Use the specified maximum size of each trace file. By default, the number entered is treated as bytes. Alternatively, you can include a suffix to the number to indicate kilobytes (KB), megabytes (MB), or gigabytes (GB). If you specify a maximum file size, you also must specify a maximum number of trace files with the **files** option.

Range: 10 KB through 1 GB

Default: 128 KB

world-readable—(Optional) Enable unrestricted file access.

no-world-readable—(Default) Disable unrestricted file access. This means the log file can be accessed only by the user who configured the tracing operation.

flag *flag*—Use the specified tracing operation. To specify more than one tracing operation, include multiple **flag** statements. You can include the following flags:

- **all**—Trace all operations.
- **configuration**—Trace configuration events.
- **error**—Trace events related to catastrophic errors in daemon.
- **ipc**—Trace IPC events.
- **ppm**—Trace ppm events.
- **statistics**—Trace statistics.

Required Privilege Level

trace—To view this statement in the configuration.
trace-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Tracing RPM Operations on MX, M, T and ACX Series Routers | 633

transfer

Syntax

```
transfer {
    record-level number;
    timeout seconds;
}
```

Hierarchy Level

```
[edit services flow-collector file-specification variant variant-number]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify when to send the flow collection file. The file is sent when either of the two conditions is met.

Options

record-level *number*—Use the specified number of flow collection files collected.

timeout *seconds*—Use the specified timeout duration.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring File Formats | 272

transfer-log-archive

Syntax

```
transfer-log-archive {
    archive-sites {
        ftp:url {
            password "password";
            username username;
        }
    }
    filename-prefix prefix;
    maximum-age minutes;
}
```

Hierarchy Level

```
[edit services flow-collector]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure the filename prefix, maximum age, and destination FTP server for log files containing the transfer activity history for a flow collector interface.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

traps

Syntax

```
traps traps;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name],
[edit services rpm twamp client control-connection control-client-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Support at the [**edit services rpm twamp client control-connection *control-client-name***] hierarchy level introduced in Junos OS Release 15.1 for MX Series routers.

Description

Set the trap bit to generate traps for probes. Traps are sent if the configured threshold is met or exceeded.

Options

traps—Specify one or more traps. The following options are supported:

- **control-connection-closed**—Generate traps when the control connection is closed.
- **egress-jitter-exceeded**—Generate traps when the jitter in egress time threshold is met or exceeded.
- **egress-std-dev-exceeded**—Generate traps when the egress time standard deviation threshold is met or exceeded.
- **egress-time-exceeded**—Generate traps when the maximum egress time threshold is met or exceeded.
- **ingress-jitter-exceeded**—Generate traps when the jitter in ingress time threshold is met or exceeded.
- **ingress-std-dev-exceeded**—Generate traps when the ingress time standard deviation threshold is met or exceeded.
- **ingress-time-exceeded**—Generate traps when the maximum ingress time threshold is met or exceeded.
- **jitter-exceeded**—Generate traps when the jitter in round-trip time threshold is met or exceeded.
- **probe-failure**—Generate traps when successive probe loss thresholds are crossed.
- **rtt-exceeded**—Generate traps when the maximum round-trip time threshold is met or exceeded.
- **std-dev-exceeded**—Generate traps when the round-trip time standard deviation threshold is met or exceeded.

- **test-completion**—Generate traps when a test is completed.
- **test-failure**—Generate traps when the total probe loss threshold is met or exceeded.
- **test-iteration-done**—Generate traps when all test sessions under control connections complete one test iteration.

> **NOTE:** For RPM traps to be generated, you must configure the **remote-operations** SNMP trap category by including the **categories** statement at the **[edit snmp trap-group *trap-group-name*]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

Understanding Two-Way Active Measurement Protocol on Routers | **582**

ttl

Syntax

```
ttl hops;
```

Hierarchy Level

```
[edit services dynamic-flow-capture capture-group client-name content-destination identifier]
```

Release Information

Statement introduced in Junos OS Release 7.4.

Description

Configure the time-to-live (TTL) value for the IP-IP header.

Options

hops—TTL value.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring the Content Destination | 323

ttl (RPM probe)

Syntax

```
ttl hop-count;
```

Hierarchy Level

```
[edit services rpm probe owner test test-name ttl hop-count]
```

Release Information
Statement introduced in Junos OS Release 18.2R1 for MX Series routers.

Description
Specify the maximum hop count (TTL) for all types of probes (IPv4 and IPv6) in real-time performance monitoring (RPM) and Two-Way Active Management Protocol (TWAMP). This can be useful when it is necessary to restrict the scope of the RPM probes so a probes does not unintentionally monitor an alternative path to the destination (such as may occur after a BGP re-routing). Another example is to monitor direct reachability by specifying a TTL of 1. Probes that exceed the number set for TTL are discarded.

The TTL configuration is supported on Routing Engine-based RPM, Multiservices Modular PIC Concentrator (MS-MPC) and Multiservices Modular Interfaces Card (MS-MIC)-based RPM, and Two-Way Active Management Protocol (TWAMP).

You can set a TTL value for the probes listed below.

Software time stamping

- icmp-ping, and icmp-ping time stamping
- icmp6-ping
- udp-ping, and udp-ping time stamping
- tcp-ping
- http-get, and http-metadata-get
- BGP neighbor monitoring using TCP/UDP

Hardware time stamping

- icmp-ping and icmp-ping-timestamp
- udp-ping and udp-ping-timestamp

MS-MPC-PIC based probes (delegate)

- icmp-ping, icmp-ping-timestamp, and icmp6-ping,

MS-MPC-PIC hardware timestamp

- icmp-ping and icmp-ping-timestamp
- udp-ping and udp-ping-timestamp

TWAMP probe

- inline TWAMP client

Options

hop-count—Prior to Junos OS Release 18.2R1, for RPM, the RPM client always sent a TTL of 64 to the RPM server under the IPv4 or IPv6 header. For TWAMP clients, the TTL was 255 sent in the IPv4 header. In Junos OS Release 18.2R1 and later, you can specify the TTL you want for RPM and TWAMP probes.

Range: 1 through 255 for both RPM and TWAMP

Default: 64

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Probes on M, MX and T Series Routers and EX Series Switches | **591**

tunnel-observation

Syntax

```
tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
```

Hierarchy Level

```
[edit services flow-monitoring version9 template template-name]
[edit services flow-monitoring version-ipfix template template-name]
```

Release Information

Statement introduced in Junos OS Release 18.1R1 on PTX Series routers.
ipv4 and **ipv6** options added in Junos OS Release 18.2R1.
Statement introduced in Junos OS Release 18.4R1 on MX Series routers.

Description

Specify the types of MPLS flows on which to enable inline flow monitoring. If you do not configure a **tunnel-observation** type, then plain MPLS flow records are created.

Options

ipv4—Enable flow monitoring for MPLS-IPv4 traffic. You must also configure **mpls-template** at the **[edit services flow-monitoring (version9 | version-ipfix) template *template-name*]** hierarchy level.

ipv6—Enable flow monitoring for MPLS-IPv6 traffic. You must also configure **mpls-template** at the **[edit services flow-monitoring (version9 | version-ipfix) template *template-name*]** hierarchy level. If you are running inline flow monitoring on a Lookup (LU) card on an MX Series router, you must also configure **use-extended-flow-memory** at the **[edit chassis fpc *slot-number* inline-services]** hierarchy level to create MPLS-IPv6 flow records.

mpls-over-udp—(PTX Series only) Enable flow monitoring for MPLS-over-UDP traffic. Monitoring looks past the tunnel header to report the inner payload of the packets. For an MPLS-over-UDP flow that is carried between IPv4 endpoints, you must also configure **ipv4-template** at the **[edit services flow-monitoring (version9 | version-ipfix) template *template-name*]** hierarchy level. For an MPLS-over-UDP flow that is encapsulated in an RSVP-TE LSP, you must also configure **mpls-ipvx-template** in Junos OS Release 18.1 or **mpls-template** starting in Junos OS 18.2R1 at the **[edit services flow-monitoring (version 9 | version-ipfix) template *template-name*]** hierarchy level.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring on PTX Series Routers | 458

Inline Active Flow Monitoring of MPLS-over-UDP Flows on PTX Series Routers | 465

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

twamp

Syntax

```
twamp {
    server {
        authentication-mode mode;
        authentication-key-chain identifier {
            key-id identifier {
                secret password-string;
            }
        }
        client-list list-name {
            [ address address ];
        }
        inactivity-timeout seconds;
        max-connection-duration hours;
        maximum-connections count;
        maximum-connections-per-client count;
        maximum-sessions count;
        maximum-sessions-per-connection count;
        port number;
        routing-instance-list {
            instance-name {
                port number;
            }
        }
        server-inactivity-timeout minutes;
        tcp-keepcnt
        tcp-keepidle
        tcp-keepintvl
    }
}
```

Hierarchy Level

```
[edit services rpm]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Configure the Two-Way Active Measurement Protocol (TWAMP) responder or sever settings on all M Series and T Series routers that support Multiservices PICs (running in either Layer 2 or Layer 3 mode), and on MX Series routers.

TWAMP is an open protocol for measurement of two-way metrics. The host that initiates the TCP connection takes the roles of the control-client and (in the two-host implementation) the session-sender. Such a device is also called the TWAMP client. The host that acknowledges the TCP connection accepts the roles of a server and (in the two-host implementation) and the session-reflector. Such a device is also called the TWAMP server. The TWAMP-Test messages are exchanged between the session-sender and the session-reflector, and the TWAMP-Control messages are exchanged between the control-client and the server.

The following addresses cannot be used for the **client-list** source IP address used for probes:

- 0.0.0.0
- 127.0.0.0/8 (loopback)
- 224.0.0.0/4 (multicast)
- 255.255.255.255 (broadcast)

The remaining statements are described separately.

Required Privilege Level
system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | **610**

twamp-server

Syntax

```
twamp-server;
```

Hierarchy Level

```
[edit interfaces sp-fpc/pic/port unit logical-unit-number]
```

Release Information

Statement introduced in Junos OS Release 9.3.

Description

Specify the service PIC logical interface to provide the TWAMP service.

Required Privilege Level

system—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring TWAMP on MX, M and T Series Routers and EX4300 Series Switches | 610

trio-flow-offload

Syntax

```
trio-flow-offload minimum-bytes minimum-bytes;
```

Hierarchy Level

```
[edit interfaces interface-name services-options]
```

Release Information

Statement introduced in Junos OS Release 12.1.

Description

Enable any plug-in or daemon on a PIC to generate a request to off-load flows to the Packet Forwarding Engine. This command is available on MX Series routers with Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs).

> **NOTE:** This feature is not supported for Broadband Edge subscribers (given that service PIC off load is not available with aggregate Ethernet (AE).

Options

minimum-bytes—Minimum number of bytes that trigger offloading. When this option is omitted, offloading is triggered when both the forward and reverse flows of the session have begun, meaning that at least one packet has flowed in each direction.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Offloading on MX Series Routers | 71

udp

Syntax

```
udp {
   destination-interface interface-name;
   port port;
}
```

Hierarchy Level

```
[edit services rpm probe-server]
```

Release Information

Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 9.3 for EX Series switches.
Statement introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Statement introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Specify the port information for the UDP server.

The remaining statements are explained separately. See CLI Explorer.

> **NOTE:** The **destination-interface** statement is not supported on PTX Series routers.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring RPM Receiver Servers | 601

udp-tcp-port-swap (RFC 2544 Benchmarking)

Syntax

```
udp-tcp-port-swap;
```

Hierarchy Level

[edit **services** rpm **rfc2544-benchmarkingtests test-name** *test-name*]

Release Information

Statement introduced in Junos OS Release 12.3X53 for ACX Series routers.
Statement introduced in Junos OS Release 14.2 for MX104 Universal Routing Platforms.

Description

Swap source and destination UDP ports in the test packets. Only UDP port swap and UDP over IPv4 traffic is supported.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- rfc2544-benchmarking | **1108**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- Configuring an RFC 2544-Based Benchmarking Test | **659**

unit

Syntax

```
unit logical-unit-number {
    family inet {
        address address {
            destination destination-address;
        }
        filter {
            group filter-group-number;
            input filter-name;
            output filter-name;
        }
        sampling direction;
    }
}
```

Hierarchy Level

```
[edit interfaces interface-name]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.

Options

logical-unit-number—Number of the logical unit.

Range: 0 through 16,384

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Junos OS Network Interfaces Library for Routing Devices for other statements that do no affect services interfaces.

Junos OS Network Interfaces Library for Routing Devices

use-extended-flow-memory

Syntax

```
use-extended-flow-memory;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services]
```

Release Information

Statement introduced in Junos OS Release 16.1 for MX Series routers.

Description

Configure the service to extended flow memory. This service provides more scale in flows for inline services sampling.

The new configuration **set chassis fpc slot slot-number inline-services use-extended-flow-memory** allows you to configure table to operate in side band mode with side band memory. This configuration is applicable only on a Lookup (LU) platform. It is not applicable for XL line card because XL has dedicated DMEM memory to hold 64M flow entries.

If you are configuring inline flow monitoring of MPLS-IPv6 flows on an LU platform, you must configure **use-extended-flow-memory** to get MPLS-IPv6 flow records. If you do not configure **use-extended-flow-memory** on an LU platform, plain MPLS flow records are created.

NOTE: This configuration is supported only on LU platforms.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

Including Fragmentation Identifier and IPv6 Extension Header Elements in IPFIX Templates on MX Series Routers | 536

username (Services)

Syntax

```
username user-name;
```

Hierarchy Level

```
[edit services flow-collector transfer-log-archive archive-sites]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the username for the transfer log server.

Options

user-name—FTP server username.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Transfer Logs | 273

variant

Syntax

```
variant variant-number {
    data-format format;
    name-format format;
    transfer {
        record-level number;
        timeout seconds;
    }
}
```

Hierarchy Level

```
[edit services flow-collector file-specification]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Configure a variant of the file format.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring File Formats | 272

version

Syntax

```
version format;
```

Hierarchy Level

```
[edit forwarding-options accounting name output flow-server hostname],
[edit forwarding-options sampling instance instance-name family (inet |inet6 |mpls) output flow-server hostname],
[edit forwarding-options sampling family (inet |inet6 |mpls) output flow-server hostname]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Specify the version format of the aggregated flows exported to a cflowd server.

Options

format—Format of the flows.

Values: 5 or 8

Default: 5

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

export-format | 936

Enabling Flow Aggregation | 489

version (Flow Monitoring Logs for NAT)

Syntax

```
version (ipfix | v9);
```

Hierarchy Level

```
[edit services jflow-log template-profile template–profile-name]
```

Release Information

Statement introduced in Junos OS Release 15.1.

Description

Specify the flow template format, such as IPFIX or version 9, to be used for generating flow monitoring records for NAT events and for transmitting them to the collector.

Options

ipfix—Use the IPFIX flow template format for flow monitoring logs for NAT events.

v9—Use the version 9 flow template format for flow monitoring logs for NAT events.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

- Understanding NAT Event Logging in Flow Monitoring Format on an MX Series Router or NFX250 | **285**
- Configuring Log Generation of NAT Events in Flow Monitoring Record Format on an MX Series Router or NFX250 | **295**
- Monitoring NAT Events on MX Series Routers by Logging NAT Operations in Flow Template Formats | **307**
- Example: Configuring Logs in Flow Monitoring Format for NAT Events on MX Series Routers for Troubleshooting | **309**

version9 (Forwarding Options)

Syntax

```
version9 {
    template template-name;
}
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (bridge | inet |inet6 |mpls vpls) output flow-server
    hostname],
[edit forwarding-options sampling family (inet |inet6 |mpls| vpls |bridge) output flow-server hostname]
```

Release Information

Statement introduced in Junos OS Release 8.3.
Statement introduced in Junos OS Release 17.2R1 for PTX Series routers with third-generation FPCs installed.
Support at the following hierarchy levels introduced in Junos OS Release 18.2R1: **[edit forwarding-options sampling instance instance-name family bridge]**, **[edit forwarding-options sampling instance instance-name family vpls]**, **[edit forwarding-options sampling family bridge]**, and **[edit forwarding-options sampling family vpls]**.
Statement introduced in Junos OS Release 19.2R1 for MX Series routers with MPC10E-15C-MRATE line card to define a flow record template suitable for IPv4 or IPv6 traffic only .

Description

Specify flow monitoring version 9 properties to apply to output sampling records.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | 495

version9 (Flow Monitoring)

Syntax

```
version9 {
    template template-name {
        flow-active-timeout seconds;
        flow-inactive-timeout seconds;
        flow-key {
            flow-direction;
            vlan-id;
            output-interface;
        }
        (ipv4-template | ipv6-template | mpls-template label-position [ positions ] | mpls-ipv4-template label-position
            [ positions ] | mpls-ipvx-template);
        option-refresh-rate packets packets seconds seconds;
        options-template-id
        peer-as-billing-template;
        source-id
        template-id
        template-refresh-rate packets packets seconds seconds;
        tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
    }
}
```

Hierarchy Level

```
[edit services flow-monitoring]
```

Release Information

Statement introduced in Junos OS Release 8.3.
Statement introduced in Junos OS Release 17.2R1 for PTX Series routers with third-generation FPCs installed.
Statement introduced in Junos OS Release 19.2R1 for MX Series routers with MPC10E-15C-MRATE line card to define a flow record template suitable for IPv4 or IPv6 traffic only .

Description

Specify the version 9 output template properties to support flow monitoring.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Flow Aggregation on MX, M, vMX and T Series Routers and NFX250 to Use Version 9 Flow Templates | **495**

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | **63**

version-ipfix (Forwarding Options)

Syntax

```
version-ipfix {
    template template-name;
}
```

Hierarchy Level

```
[edit forwarding-options sampling instance instance-name family (inet | inet6 | mpls | vpls) output flow-server
    hostname]
```

Release Information

Statement introduced in Junos OS Release 10.2.
Statement introduced in Junos OS Release 12.R3 for EX Series switches.
Statement introduced in Junos OS Release 15.1F4 for PTX Series routers with third-generation FPCs installed.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Specify flow monitoring version IPFIX properties to apply to output sampling records.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | 513

version-ipfix (Services)

Syntax

```
version-ipfix {
    template template-name {
        flow-active-timeout seconds;
        flow-inactive-timeout seconds;
        flow-key {
            flow-direction;
            vlan-id;
            output-interface;
        }
        (ipv4-template | ipv6-template | mpls-ipv4-template | mpls-ipvx-template | vpls-template);
        nexthop-learning (enable |disable);
        observation-domain-id
        option-refresh-rate packets packets seconds seconds;
        options-template-id
        template-id
        template-refresh-rate packets packets seconds seconds;
        tunnel-observation [ipv4 | ipv6 | mpls-over-udp];
    }
}
```

Hierarchy Level

```
[edit services flow-monitoring]
```

Release Information

Statement introduced in Junos OS Release 10.2.
Statement introduced in Junos OS Release 12.R3 for EX Series switches.
Statement introduced in Junos OS Release 15.1F4 for PTX Series routers with third-generation FPCs installed.
Statement introduced in Junos OS Release 17.2R1 for QFX10002 switches.
Statement introduced in Junos OS Release 17.4R1 for QFX10008 and QFX10016 switches.

Description

Specify the IPFIX output template properties to support flow monitoring.

The remaining statements are explained separately. See CLI Explorer.

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

Configuring Flow Aggregation to Use IPFIX Flow Templates on MX, vMX and T Series Routers, EX Series Switches and NFX250, and SRX Devices | 513

Configuring Inline Active Flow Monitoring on PTX Series Routers | 458

video-monitoring

Syntax

```
video-monitoring {
    interfaces {
        interface-name {
            family {
                inet {
                    input-flows {
                        input-flow-name {
                            destination-address [ address ];
                            destination-port [ port ];
                            source-address [ address ];
                            source-port [ port ];
                            template template-name;
                        }
                    }
                    output-flows {
                        output-flow-name {
                            destination-address [ address ];
                            destination-port [ port ];
                            source-address [ address ];
                            source-port [ port ];
                            template template-name;
                        }
                    }
                }
                inet6 {
                    input-flows {
                        input-flow-name {
                            destination-address [ address ];
                            destination-port [ port ];
                            source-address [ address ];
                            source-port [ port ];
                            template template-name;
                        }
                    }
                    output-flows {
                        output-flow-name {
                            destination-address [ address ];
                            destination-port [ port ];
                            source-address [ address ];
                            source-port [ port ];
                            template template-name;
```

```
            }
          }
        }
        mpls {
          input-flows {
            input-flow-name {
              (destination-address [ address ] | source-address [ address ]);
              destination-port [ port ];
              payload-type (ipv4 | ipv6);
              source-port [ port ];
              template template-name;
            }
          }
          output-flows {
            output-flow-name {
              (destination-address [ address ] | source-address [ address ]);
              destination-port [ port ];
              payload-type (ipv4 | ipv6);
              source-port [ port ];
              template template-name;
            }
          }
        }
      }
    }
}
```

```
templates {
    template-name {
        interval-duration interval-duration;
        inactive-timeout inactive-timeout;
        rate {
            (layer3 layer3-packets-per-second | media media-bits-per-second);
        }
        delay-factor {
            disable;
            threshold {
                (info | warning | critical) delay-factor-threshold;
            }
        }
        media-loss-rate {
            disable;
            threshold {
                (info | warning | critical) percentage mlr-percentage | packet-count mlr-packet-count);
            }
        }
        media-rate-variation {
            ;
            threshold {
                (info |warning | critical) mrv-variation;
            }
        }
        media-packets-count-in-layer3 media-packets-count-in-layer3;
        media-packet-size media-packet-size;
    }
  }
}
```

Hierarchy Level

```
[edit services]
```

Release Information

Statement introduced in Junos OS Release 14.1.

Description

Define the options for video monitoring using media delivery index options for metrics.

The remaining statements are explained separately.

Required Privilege Level

interface—To view this statement in the configuration.

interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Video Monitoring on MX Series Routers | 833

vpls-flow-table-size

Syntax

```
vpls-flow-table-size units;
```

Hierarchy Level

```
[edit chassis fpc slot-number inline-services flow-table-size]
```

Release Information

Statement introduced in Junos OS Release 13.2.

Description

Configure the size of the VPLS flow table in units of 256K entries.

> **NOTE:** Any change in the configured size of the flow table size initiates an automatic reboot of the FPC.

> **NOTE:** Starting with Junos OS Release 17.3R1, the maximum number of 256K flow entries that you can configure for IPv4 flow tables and IPv6 flow tables is 256 on MPC5Es and MPC6Es with with 4 GB DDR3 memory or higher. The maximum number of 256K flow entries that you can configure for IPv4 flow tables and IPv6 flow tables is 245 on MPC5Es and MPC6Es with DDR3 memory lower than 4 GB.

> **NOTE:** The recommended flow table size is 4 so that it can scale up to 4x256K flows, which is 1M. You can configure more, however, the system will issue a warning message.

Options

units—Number of 256K flow entries available for the VPLS flow table.

Range: 1 through 245

Default: 15 (3840K)

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63

vpls-template

Syntax

```
vpls-template;
```

Hierarchy Level

[edit **services flow-monitoring version-ipfix template**]

Release Information
Statement introduced in Junos OS Release 13 .2.

Description
Specify that the IPFIX template is used only for VPLS records. Starting in Junos OS Release 18.2R1, the **vpls-template** option is deprecated; use the **bridge-template** option instead. The **bridge-template** option supports both VPLS and bridge records.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Inline Active Flow Monitoring Using Routers, Switches or NFX250 | 63
bridge-template | 882

world-readable

Syntax

```
(world-readable | no-world-readable);
```

Hierarchy Level

```
[edit forwarding-options port-mirroring traceoptions file],
[edit forwarding-options sampling family (inet |inet6 |mpls) output file],
[edit forwarding-options sampling traceoptionsfile]
```

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Enable unrestricted file access.

Options

no-world-readable—Restrict file access to owner. This is the default.

world-readable—Enable unrestricted file access.

Default: no-world-readable

Required Privilege Level

interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

RELATED DOCUMENTATION

Configuring Port Mirroring on M, T MX, ACX, and PTX Series Routers | 547

Configuring Traffic Sampling on MX, M and T Series Routers | 375

CHAPTER 20

Operational Commands

IN THIS CHAPTER

clear passive-monitoring statistics

Syntax

```
clear passive-monitoring statistics (all | interface interface-name)
```

Release Information
Command introduced in Junos OS Release 7.6.

Description
(M40e, M160, and M320 Series routers and T Series routers only) Clear statistics for one passive monitoring interface or for all passive monitoring interfaces.

Options
all—Clear statistics for all configured passive monitoring interfaces.

interface *interface-name*—Clear statistics for the specified passive monitoring interface (**mo-*fpc*/*pic*/*port***).

Required Privilege Level
network

List of Sample Output
clear passive-monitoring statistics on page 1230

Output Fields
When you enter this command, you are not provided feedback on the status of your request.

Sample Output

clear passive-monitoring statistics

```
user@host> clear passive-monitoring statistics interface mo-5/0/0
```

clear services accounting statistics inline-jflow

Syntax

```
clear services accounting statistics inline-jflow
<inline-jflow (fpc-slot slot-number)>
```

Release Information

Command introduced in Junos OS Release 14.2 for MX Series routers.

Description

Clear inline flow statistics for a specified FPC.

Options

fpc-slot *slot-number*—Clear inline flow statistics for the specified FPC.

- MX80 Series routers only—Replace ***slot-number*** with a value from **0** through **1**.
- MX104 Series routers only—Replace ***slot-number*** with a value from **0** through **2**.
- MX240 Series routers only—Replace ***slot-number*** with a value from **0** through **2**.
- MX480 Series routers only—Replace ***slot-number*** with a value from **0** through **5**.
- MX960 Series routers only—Replace ***slot-number*** with a value from **0** through **11**.
- MX2010 Series routers only—Replace ***slot-number*** with a value from **0** through **9**.
- MX2020 Series routers only—Replace ***slot-number*** with a value from **0** through **19**.

Required Privilege Level

view

RELATED DOCUMENTATION

show services accounting flow | **1299**

List of Sample Output

clear services accounting statistics inline-jflow on page 1232

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear services accounting statistics inline-jflow

```
user@host> run clear services accounting statistics inline-jflow fpc-slot 5
```

```
Statistics Cleared
```

clear services dynamic-flow-capture

Syntax

```
clear services dynamic-flow-capture capture-group group-name
<criteria-identifier identifier>
<destination-identifier identifier>
<force>
<static>
```

Release Information
Command introduced in Junos OS Release 7.4.

Description
(M320 Series routers and T Series routers only) Clear dynamic flow capture information for specified capture group.

Options
capture-group *group-name*—Use the specified capture-group identifier.

criteria-identifier *identifier*—(Optional) Use the specified criteria identifier.

destination-identifier *identifier*—(Optional) Use the specified content destination identifier.

force—(Optional) Force clearing of criteria.

static—(Optional) Clear static criteria.

Required Privilege Level
network

List of Sample Output
clear services dynamic-flow-capture on page 1233

Output Fields
When you enter this command, you are not provided feedback on the status of your request.

Sample Output

clear services dynamic-flow-capture

```
user@host> clear services dynamic-flow-capture capture-group flow-a
```

clear services flow-collector statistics

Syntax

```
clear services flow-collector statistics (all | interface interface-name)
```

Release Information
Command introduced before Junos OS Release 7.4.

Description
(M40e, M160, and M320 Series routers and T Series routers only) Clear statistics for one flow collector interface or for all flow collector interfaces.

Options
all—Clear statistics for all configured flow collector interfaces.

interface *interface-name*—Clear statistics for the specified flow collector interface (**cp-*fpc*/*pic*/*port***).

Required Privilege Level
network

List of Sample Output
clear services flow-collector statistics on page 1234

Output Fields
When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear services flow-collector statistics

`user@host>` **clear services flow-collector statistics interface cp-5/0/0**

```
Flow collector interface: cp-5/0/0
Interface state: Collecting flows
Statistics cleared successfully
```

clear services inline-monitoring statistics

Syntax

```
clear services inline-monitoring statistics fpc-slot fpc-slot
collector-name collector-name
instance-name instance-name
```

Release Information

Command introduced in Junos OS Release 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Clear statistics for inline monitoring services.

Options

collector-name *collector-name*—Clear collector level statistics.

fpc-slot *fpc-slot*—Clear statistics for the specified FPC slot.

Range: 0 through 11

instance-name *instance-name*—Clear instance level statistics.

Required Privilege Level

view

RELATED DOCUMENTATION

show services inline-monitoring statistics fpc-slot | 1348

Understanding Inline Monitoring Services | 362

clear services rpm twamp server connection

Syntax

clear services rpm twamp server connection
<connection-id>

Release Information
Command introduced in Junos OS Release 9.3.

Description
Clear connections established between the real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) server and control clients. By default all established connections are cleared (along with the sessions on those connections). To clear only a specific connection, specify the connection ID when you issue the command.

Options
connection-id—(Optional) Specific connection to clear.

Required Privilege Level
clear

clear services service-sets statistics jflow-log

Syntax

```
clear services service-sets statistics jflow-log
<service-set service-set-name>
<interface interface-name>
```

Release Information

Command introduced in Junos OS Release 15.1.

Description

Clear flow monitoring log statistics for the logs generated in IPFIX or version 9 format for one services interface or for all services interfaces, and for one named service set or all service sets on the interface or interfaces.

Options

none—Clear flow monitoring log for all configured services interfaces and their service sets.

interface *interface-name*—(Optional) Clear flow monitoring log statistics for the specified services interface. On M Series, MX Series, and T Series routers, the ***interface-name*** can be **ms-*fpc/pic/port***. It is supported only on MS-MICs and MS-MPCS.

service-set *service-set-name*—(Optional) Clear flow monitoring log statistics for the specified services interface.

Required Privilege Level

network

RELATED DOCUMENTATION

show services service-sets statistics syslog

List of Sample Output

clear services service-sets statistics jflow-log interface on page 1238

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear services service-sets statistics jflow-log interface

user@host> **clear services service-sets statistics jflow-log interface ms-5/0/0**

```
Interface: ms-5/0/0
```

clear services video-monitoring mdi errors fpc-slot

Syntax

```
clear services video-monitoring mdi errors <fpc-slot fpc-slot>
```

Release Information
Command introduced in Junos OS Release 14.1.

Description
Clear all media delivery index error counters for the specified FPC slot or for all FPC slots.

Options
none—Clear error counters for all FPC slots.

fpc-slot—(Optional) Clear error counters for the specified FPC slot.

Required Privilege Level
clear

RELATED DOCUMENTATION

show services video-monitoring mdi stats fpc-slot | 1437

List of Sample Output
clear services video-monitoring mdi errors on page 1239

Output Fields
When you enter this command, you are provided feedback on the status of your request.

Sample Output

clear services video-monitoring mdi errors

`user@host>` **clear services video-monitoring mdi errors**

```
Errors counters cleared
```

clear services video-monitoring mdi statistics fpc-slot

Syntax

```
clear services video-monitoring mdi statistics fpc-slot fpc-slot
```

Release Information

Command introduced in Junos OS Release 14.1.

Description

Clear all media delivery index statistics counters except for active flows.

Options

fpc-slot—Number of the FPC slot.

Required Privilege Level

clear

RELATED DOCUMENTATION

show services video-monitoring mdi stats fpc-slot | **1437**

request services flow-collector change-destination primary interface

Syntax

```
request services flow-collector change-destination primary interface cp-fpc/pic/port
<clear-files>
<clear-logs>
<immediately | gracefully>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Switch to the primary File Transfer Protocol (FTP) server that is configured as a flow collector.

Options

none—Switch to the primary FTP server.

cp-*fpc/pic/port*—Use the specified flow collector interface name for the primary destination.

clear-files—(Optional) Request clearing of existing data files in the FTP wait queue when the switch takes place.

clear-logs—(Optional) Request clearing of existing logs when the switch takes place.

immediately | gracefully—(Optional) Specify whether you want the switch to take place immediately, or to affect only newly created files.

Required Privilege Level

maintenance

List of Sample Output

request services flow-collector change-destination primary interface on page 1241

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

request services flow-collector change-destination primary interface

```
user@host> request services flow-collector change-destination primary interface cp-6/0/0
```

```
Flow collector interface: cp-6/0/0
Interface state: Collecting flows
Destination change successful
```

request services flow-collector change-destination secondary interface

Syntax

```
request services flow-collector change-destination secondary interface cp-fpc/pic/port
<clear-files>
<clear-logs>
<immediately | gracefully>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Switch to the secondary File Transfer Protocol (FTP) server that is configured as a flow collector.

Options

none—Switch to the secondary FTP server.

cp-*fpc/pic/port*—Use the specified flow collector interface name (**cp-*fpc/pic/port***) for the secondary destination.

clear-files—(Optional) Request clearing of existing data files in the FTP wait queue when the switch takes place.

clear-logs—(Optional) Request clearing of existing logs when the switch takes place.

immediately | gracefully—(Optional) Specify whether you want the switch to take place immediately, or to affect only newly created files.

Required Privilege Level

maintenance

List of Sample Output

request services flow-collector change-destination secondary interface on page 1243

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

request services flow-collector change-destination secondary interface

```
user@host> request services flow-collector change-destination secondary interface cp-6/0/0
```

```
Flow collector interface: cp-6/0/0
Interface state: Collecting flows
Destination change successful
```

request services flow-collector test-file-transfer

Syntax

request services flow-collector test-file-transfer *filename* interface (all | cp-*fpc*/*pic*/*port*) (channel-zero | channel-one) (primary | secondary)

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers, PTX Series, and T Series routers only) Transfer a test file to the primary or secondary File Transfer Protocol (FTP) server that is configured as a flow collector. This command verifies that the output side of the flow collector interface is operating properly.

Options

filename—Name of the test file to transfer.

interface (all | cp-*fpc*/*pic*/*port*)—Transfer a test file of flows from all configured flow collector interfaces or from only the specified interface.

channel-zero | channel-one—Transfer a file from export channel 0 (unit 0) or channel 1 (unit 1) of the PIC.

primary | secondary—Transfer a file to the primary or secondary server configured as a flow collector.

Required Privilege Level

network

List of Sample Output

request services flow-collector test-file-transfer interface channel-one primary on page 1245

Output Fields

When you enter this command, you are provided feedback on the status of your request.

Sample Output

request services flow-collector test-file-transfer interface channel-one primary

`user@host>` **request services flow-collector test-file-transfer test_file interface cp-7/1/0 channel-one primary**

```
Flow collector interface: cp-7/1/0
Interface state: Collecting flows
Response: Test file transfer successfully scheduled
```

request services rpm twamp

Syntax

```
request services rpm twamp (start | stop) client <control-connection-name>
```

Release Information

Command introduced in Junos OS Release 15.1 for MX Series routers.

Description

Start or stop a TWAMP session. You can start or stop all of the sessions for all of the TWAMP clients, or start or stop a session for a specific TWAMP client. When you start all the test session configured for a particular TWAMP client, the control-client initiates all requested testing with a Start-Sessions message, and the server sends an acknowledgment. If the control connection is not active between the server and the client, the control connection is also established and the test connections are started later. If the control-client name is not specified, all the configured test sessions are commenced.

When you stop the test session, the control connection is closed only after the Stop-sessions message is sent from the TWAMP client to the TWAMP server. If the control-client name is not specified, all the configured test sessions are closed.

Options

start client—Start the TWAMP session between the TWAMP client and the TWAMP server.

stop client—Terminate the TWAMP session between the TWAMP client and the TWAMP server.

control-connection-name—(Optional) Start or stop the TWAMP session with the server only for the specified control-connection or TWAMP control-client.

Required Privilege Level

view

List of Sample Output

request services rpm twamp start client on page 1247

Output Fields

When you enter this command, you are not provided feedback on the status of your request.

Sample Output

request services rpm twamp start client

```
user@host> request services rpm twamp start client c1
```

show forwarding-options next-hop-group

Syntax

```
show forwarding-options next-hop-group
<terse | brief | detail>
<group-name>
```

Release Information

Command introduced in Junos OS Release 9.6.
Command introduced in Junos OS Release 12.3R2 for EX Series switches.
Support for IPv6 introduced in Junos OS Release 14.2 for the MX Series routers.

Description

Display current state of next-hop groups.

Options

terse | brief | detail—(Optional) Display the specified level of output.

group-name—(Optional) Display a single next-hop group.

Required Privilege Level

view

RELATED DOCUMENTATION

show forwarding-options port-mirroring | 1252

List of Sample Output

show forwarding-options next-hop-group terse on page 1249
show forwarding-options next-hop-group brief on page 1250
show forwarding-options next-hop-group detail on page 1250

Output Fields

Table 112 on page 1248 lists the output fields for the **show forwarding-options next-hop-group** command. Output fields are listed in the approximate order in which they appear.

Table 112: show forwarding-options next-hop-group Output Fields

Field Name	Field Description	Level of Output
Next-hop-group	Name of next-hop group.	All levels

Table 112: show forwarding-options next-hop-group Output Fields *(continued)*

Field Name	Field Description	Level of Output
Type	Next-hop group type, such as **inet**, **inet6** or **layer-2**.	All levels
State	Next-hop group state, either **up** or **down**.	All levels
Members Interfaces	Names of interfaces to which next-hop group members belong.	**brief detail**
Member Subgroup	Names of subgroups to which next-hop group members belong.	**brief detail**
Number of members configured	Number of next-hop group members configured.	**detail**
Number of members that are up	Number of next-hop group members that are up.	**detail**
Number of subgroups configured	Number of subgroups configured.	**detail**
Number of subgroups that are up	Number of subgroups that are up.	**detail**

Sample Output

show forwarding-options next-hop-group terse

`user@host>` **show forwarding-options next-hop-group terse**

```
Next-hop-group                Type            State
  nhg                           inet            up
  nhg6                          inet6           up
  vpls_nhg_2                  layer-2         down
```

show forwarding-options next-hop-group brief

user@host> **show forwarding-options next-hop-group brief**

```
Next-hop-group: nhg
      Type:  inet
      State: up
      Members Interfaces:
        ge-0/2/8.0            next-hop  192.0.2.10
        ge-5/1/8.0            next-hop  198.51.100.10
        ge-5/1/9.0            next-hop  203.0.113.10

Next-hop-group: nhg6
      Type:  inet6
      State: up
      Members Interfaces:
        ge-5/1/5.0            next-hop  2001:db8::1:10
        ge-5/1/6.0            next-hop  2001:db8::20:10        Member Subgroup:
nhsg6
          Members Interfaces:
            ge-5/0/4.0            next-hop  2001:db8::3:1
            ge-5/1/4.0            next-hop  2001:db8::4:1

Next-hop-group: vpls_nhg_2
Type: layer-2        State: down
```

show forwarding-options next-hop-group detail

user@host> **show forwarding-options next-hop-group detail**

```
Next-hop-group: nhg
Type: inet
State: up
Number of members configured    : 3
Number of members that are up   : 3
Number of subgroups configured  : 0
Number of subgroups that are up : 0
Members Interfaces:                                 State
  ge-0/2/8.0            next-hop  192.0.2.10        up
  ge-5/1/8.0            next-hop  203.0.113.10        up
  ge-5/1/9.0            next-hop  198.51.100.10.10        up

Next-hop-group: nhg6
```

```
Type: inet6
State: up
Number of members configured    : 2
Number of members that are up   : 2
Number of subgroups configured  : 1
Number of subgroups that are up : 1
Members Interfaces:                                 State
ge-5/1/5.0             next-hop  2001:db8::1:10       up
ge-5/1/6.0             next-hop  2001:db8::20:10      up
Member Subgroup: nhsg6                            up
       Number of members configured    : 2
       Number of members that are up   : 2
       Members Interfaces:                                 State
           ge-5/0/4.0             next-hop  2001:db8::3:1   up
           ge-5/1/4.0             next-hop  2001:db8::4:1   up

Next-hop-group: vpls_nhg_2
Number of members configured    : 2
Number of members that are up   : 0
Number of subgroups configured  : 0
Number of subgroups that are up : 0
Type: layer-2        State: down
Members Interfaces:       State
   ge-2/2/1.100        down
   ge-2/3/9.0        down
```

show forwarding-options port-mirroring

Syntax

```
show forwarding-options port-mirroring
<terse | detail>
<instance-name>
```

Release Information

Command introduced in Junos OS Release 9.6.
Command introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Display current state of port-mirroring instances.

Options

terse | detail—(Optional) Display the specified level of output.

instance-name—(Optional) Display a single port-mirroring instance.

Required Privilege Level

view

RELATED DOCUMENTATION

List of Sample Output

show forwarding-options port-mirroring terse on page 1253
show forwarding-options port-mirroring detail on page 1253

Output Fields

Table 113 on page 1252 lists the output fields for the **show forwarding-options port-mirroring** command. Output fields are listed in the approximate order in which they appear.

Table 113: show forwarding-options port-mirroring Output Fields

Field Name	Field Description	Level of Output
Instance Name	Name of port-mirroring instance.	All levels
Instance Id	Instance identification number.	All levels
State	Instance state, either **up** or **down**.	All levels

Table 113: show forwarding-options port-mirroring Output Fields *(continued)*

Field Name	Field Description	Level of Output
Input parameters		
Rate	Rate (ratio of packets sampled).	**detail**
Run-length	Run length (number of consecutive packets sampled).	**detail**
Maximum-packet-length	Maximum packet length.	**detail**
Output parameters		
Family	Protocol family.	**detail**
State	Instance state, either **up** or **down**.	**detail**
Destination	Destination (next-hop group name).	**detail**
Next-hop	IP address of the next hop to the destination.	**detail**

Sample Output

show forwarding-options port-mirroring terse

`user@host>` **show forwarding-options port-mirroring terse**

```
Instance Name        Instance Id    State
&global_instance       1      up
inst1         2      up
```

show forwarding-options port-mirroring detail

`user@host>` **show forwarding-options port-mirroring detail**

```
Instance Name: pm1
  Instance Id: 2
  Input parameters:
    Rate                  : 2
    Run-length            : 0
    Maximum-packet-length : 0
```

```
Output parameters:
  Family              State     Destination          Next-hop
  inet                up        ge-0/0/0.0           10.1.1.2
  inet6               up        ge-0/0/0.0           2001:db8::2
  any                 up        ge-0/0/1.0           NA
```

show interfaces (Dynamic Flow Capture)

Syntax

```
show interfaces dfc-fpc/pic/port:channel
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index snmp-index>
<statistics>
```

Release Information

Command introduced in Junos OS Release 7.4.

Description

(M320 and M120 Series routers and T Series routers only) Display status information about the specified dynamic flow capture interface.

Options

dfc-*fpc/pic/port*:*channel*—Display standard status information about the specified dynamic flow capture interface.

brief | detail | extensive | terse—(Optional) Display the specified level of output.

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces (Dynamic Flow Capture) on page 1258

Output Fields

Table 114 on page 1256 lists the output fields for the **show interfaces** (Dynamic Flow Capture) command. Output fields are listed in the approximate order in which they appear.

Table 114: Dynamic Flow Capture show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels
Enabled	Sate of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Type	Type of interface.	All levels
Link-level type	Encapsulation type used on the physical interface.	All levels
MTU	Maximum Transmit Unit (MTU). Size of the largest packet to be transmitted.	All levels
Speed	Network speed on the interface.	All levels
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Link type	Data transmission type.	All levels
Link flags	Information about the link. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	All levels
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped: *year-month-day hour:minute:second timezone* (*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)**.	**detail extensive**
Input Rate	Input rate in bits per second (bps) and packets per second (pps).	None specified
Output Rate	Output rate in bps and pps.	None specified

Table 114: Dynamic Flow Capture show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input rate**, **Output rate**—Number of bits per second (packets per second) received and transmitted on the interface. • **Input packets**, **Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Input errors	• **Errors**—Input errors on the interface. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Frames received smaller than the runt threshold. • **Giants**—Frames received larger than the giant threshold. • **Policed Discards**—Frames that the incoming packet match code discarded because the frames did not recognize them or were not of interest. Usually, this field reports protocols that the Junos OS does not support. • **Resource errors**—Sum of transmit drops.	**extensive**
Output errors	• **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly, possibly once every 10 seconds, the cable, the remote system, or the interface is malfunctioning. • **Errors**—Sum of outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet dropped by the ASIC RED mechanism. • **Resource errors**—Sum of transmit drops.	**extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none

Table 114: Dynamic Flow Capture show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Flags	Information about the logical interface; values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Input packets	Number of packets received on the logical interface.	None specified
Output packets	Number of packets transmitted on the logical interface.	None specified
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets, Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Protocol	Protocol family configured on the logical interface (such as **iso** or **inet6**).	**detail extensive** none
MTU	MTU size on the logical interface.	**detail extensive** none
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none
Addresses, Flags	Addresses associated with the logical interface and information about the address flags. Possible values are described in the "Addresses Flags" section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none

Sample Output

show interfaces (Dynamic Flow Capture)

```
user@host> show interfaces dfc-0/0/0
```

```
Physical interface: dfc-0/0/0, Enabled, Physical link is Up
 Interface index: 146, SNMP ifIndex: 36
 Type: Adaptive-Services, Link-level type: Dynamic-Flow-Capture, MTU: 9192, Speed:
 2488320kbps
 Device flags : Present Running
 Interface flags: Point-To-Point SNMP-Traps 16384
 Link type : Full-Duplex
 Link flags : None
 Last flapped : 2005-08-26 15:08:36 PDT (01:18:42 ago)
 Input rate : 0 bps (0 pps)
 Output rate : 44800440 bps (100000 pps)

Logical interface dfc-0/0/0.0 (Index 67) (SNMP ifIndex 43)
 Flags: Point-To-Point SNMP-Traps Encapsulation: Dynamic-Flow-Capture
Input packets : 74
Output packets: 132
 Protocol inet, MTU: 9192
  Flags: Receive-options, Receive-TTL-Exceeded
  Addresses, Flags: Is-Preferred Is-Primary
   Destination: 10.36.100.1, Local: 10.36.100.2

Logical interface dfc-0/0/0.1 (Index 68) (SNMP ifIndex 49)
 Flags: Point-To-Point SNMP-Traps Encapsulation: Dynamic-Flow-Capture
Input packets : 0
Output packets: 402927263
 Protocol inet, MTU: 9192
  Flags: Receive-options, Receive-TTL-Exceeded

Logical interface dfc-0/0/0.2 (Index 69) (SNMP ifIndex 50)
 Flags: Point-To-Point SNMP-Traps Encapsulation: Dynamic-Flow-Capture
Input packets : 0
Output packets: 0
 Protocol inet, MTU: 9192
  Flags: Receive-options, Receive-TTL-Exceeded

Logical interface dfc-0/0/0.16383 (Index 70) (SNMP ifIndex 44)
 Flags: Point-To-Point SNMP-Traps Encapsulation: Dynamic-Flow-Capture
Input packets : 1427
Output packets: 98
 Protocol inet, MTU: 9192
  Flags: Receive-options, Receive-TTL-Exceeded
  Addresses, Flags: Is-Preferred Is-Primary
   Destination: 10.0.0.16, Local: 10.0.0.1
```

show interfaces (Flow Collector)

Syntax

```
show interfaces cp-fpc/pic/port:channel
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index snmp-index>
<statistics>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M Series and T Series routers only) Display status information about the specified flow collector interface.

Options

cp-*fpc/pic/port*:*channel*—Display standard status information about the specified flow collector interface.

brief | detail | extensive | terse—(Optional) Display the specified level of output.

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces extensive (Flow Collector) on page 1264

Output Fields

Table 115 on page 1260 lists the output fields for the **show interfaces** (Flow Collector) command. Output fields are listed in the approximate order in which they appear.

Table 115: Flow Collector Show interfaces Output Fields

Field Name	Field Description	Level of Output
Physical Interface		

Table 115: Flow Collector Show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Physical Interface	Name of the physical interface type.	All levels
Link	Status of the link: **up** or **down**.	All levels
Enabled	State of the interface type. Possible values are described in the "Enabled Devices" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Type	Type of interface.	All levels
Link-level type	Encapsulation type used on the physical interface.	All levels
MTU	Maximum Transmit Unit (MTU). Size of the largest packet to be transmitted.	All levels
Clocking	Reference clock source of the interface.	All levels
Speed	Network speed on the interface.	All levels
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Link type	Data transmission type.	All levels
Link flags	Information about the link. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	All levels
Physical info	Information about the physical interface.	All levels
Hold-times	Current interface hold-time up and hold-time down. Value is in milliseconds.	**detail extensive** none

Table 115: Flow Collector Show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Current address	Configured MAC address.	**detail extensive** none
Hardware address	Media access control (MAC) address of the interface.	**detail extensive** none
Alternate link address	Backup link address.	**detail extensive** none
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped: *year-month-day hour:minute:second timezone* (*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)**.	**detail extensive**
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**, **Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets**, **Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Input errors	• **Errors**—Input errors on the interface. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Frames received smaller than the runt threshold. • **Giants**—Frames received larger than the giant threshold. • **Policed Discards**—Frames that the incoming packet match code discarded because the frames did not recognize them or were not of interest. Usually, this field reports protocols that Junos does not support. • **Resource errors**—Sum of transmit drops.	**extensive**

Table 115: Flow Collector Show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Output errors	• **Carrier transitions** —Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly, possibly once every 10 seconds, the cable, the remote system, or the interface is malfunctioning. • **Errors**—Sum of outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet dropped by the ASIC RED mechanism. • **Resource errors**—Sum of transmit drops.	**extensive**
Logical Interface		
Logical interface	Name of the logical interface	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface; values are described in the “Logical Interface Flags” section under *Common Output Fields Description*.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets, Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**

Table 115: Flow Collector Show interfaces Output Fields *(continued)*

Field Name	Field Description	Level of Output
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Protocol	Protocol family configured on the logical interface (such as **iso** or **inet6**).	**detail extensive** none
MTU	MTU size on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Route table in which this address exists; for example, **Route table:0** refers to inet.0.	**detail extensive**
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none
Addresses, Flags	Information about the address flags. Possible values are described in the "Addresses Flags" section under *Common Output Fields Description*.	**detail extensive** none
Destination	IP address of the remote side of the connection.	**detail extensive** none
Local	IP address of the logical interface.	**detail extensive** none
Broadcast	Broadcast address.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**

Sample Output

show interfaces extensive (Flow Collector)

`user@host>` **show interfaces extensive cp-5/0/0**

```
Physical interface: cp-5/0/0, Enabled, Physical link is Up
  Interface index: 145, SNMP ifIndex: 52, Generation: 29
  Type: Flow-collector, Link-level type: Flow-collection, MTU: 9192,
  Clocking: Unspecified, Speed: 800mbps
  Device flags   : Present Running
```

```
Interface flags: Point-To-Point SNMP-Traps 16384
Link type      : Full-Duplex
Link flags     : None
Physical info  : Unspecified
Hold-times     : Up 0 ms, Down 0 ms
Current address: Unspecified, Hardware address: Unspecified
Alternate link address: Unspecified
Last flapped   : 2005-05-24 16:48:11 PDT (00:12:04 ago)
Statistics last cleared: Never
Traffic statistics:
 Input  bytes  :           2041661287                   0 bps
 Output bytes  :           3795049544            43816664 bps
 Input  packets:              1365534                   0 pps
 Output packets:              3865644                3670 pps
Input errors:
  Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0,
  Policed discards: 0, Resource errors: 0
Output errors:
  Carrier transitions: 2, Errors: 0, Drops: 0, MTU errors: 0,
  Resource errors: 0

Logical interface cp-5/0/0.0 (Index 74) (SNMP ifIndex 53) (Generation 28)
  Flags: Point-To-Point SNMP-Traps Encapsulation: Flow-collection
  Traffic statistics:
   Input  bytes  :           1064651568
   Output bytes  :             37144290
   Input  packets:                711324
   Output packets:                713672
  Local statistics:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Transit statistics:
   Input  bytes  :           1064651568                   0 bps
   Output bytes  :             37144290                   0 bps
   Input  packets:                711324                   0 pps
   Output packets:                713672                   0 pps
  Protocol inet, MTU: 9192, Generation: 39, Route table: 0
    Flags: Receive-options, Receive-TTL-Exceeded
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 192.0.2.2, Local: 192.0.2.1, Broadcast: Unspecified,
      Generation: 40
```

```
Logical interface cp-5/0/0.1 (Index 75) (SNMP ifIndex 54) (Generation 29)
  Flags: Point-To-Point SNMP-Traps Encapsulation: Flow-collection
  Traffic statistics:
   Input  bytes  :            976793823
   Output bytes  :             34099481
   Input  packets:               652729
   Output packets:               655127
  Local statistics:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Transit statistics:
   Input  bytes  :            976793823                    0 bps
   Output bytes  :             34099481                    0 bps
   Input  packets:               652729                    0 pps
   Output packets:               655127                    0 pps
  Protocol inet, MTU: 9192, Generation: 40, Route table: 0
    Flags: Receive-options, Receive-TTL-Exceeded
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 198.51.100.2, Local: 198.51.100.1, Broadcast: Unspecified,
      Generation: 42

Logical interface cp-5/0/0.2 (Index 80) (SNMP ifIndex 55) (Generation 30)
  Flags: Point-To-Point SNMP-Traps Encapsulation: Flow-collection
  Traffic statistics:
   Input  bytes  :                    0
   Output bytes  :           3723079376
   Input  packets:                    0
   Output packets:              2495372
  Local statistics:
   Input  bytes  :                    0
   Output bytes  :                    0
   Input  packets:                    0
   Output packets:                    0
  Transit statistics:
   Input  bytes  :                    0                    0 bps
   Output bytes  :           3723079376             43816664 bps
   Input  packets:                    0                    0 pps
   Output packets:              2495372                 3670 pps
  Protocol inet, MTU: 9192, Generation: 41, Route table: 0
    Flags: Receive-options, Receive-TTL-Exceeded
    Addresses, Flags: Is-Preferred Is-Primary
      Destination: 203.0.113.2, Local: 203.0.113.1, Broadcast: Unspecified,
```

```
        Generation: 44

  Logical interface cp-5/0/0.16383 (Index 81) (SNMP ifIndex 56) (Generation 31)
...
```

show interfaces (Flow Monitoring)

Syntax

```
show interfaces mo-fpc/pic/port:channel
<brief | detail | extensive | terse>
<descriptions>
<media>
<snmp-index snmp-index>
<statistics>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M Series and T Series routers only) Display status information about the specified flow monitoring interface.

Options

mo-*fpc/pic/port:channel*—Display standard status information about the specified flow monitoring interface.

brief | detail | extensive | terse—(Optional) Display the specified level of output.

descriptions—(Optional) Display interface description strings.

media—(Optional) Display media-specific information about network interfaces.

snmp-index *snmp-index*—(Optional) Display information for the specified SNMP index of the interface.

statistics—(Optional) Display static interface statistics.

Required Privilege Level

view

List of Sample Output

show interfaces extensive (Flow Monitoring) on page 1272

Output Fields

Table 116 on page 1268 lists the output fields for the **show interfaces** (Flow Monitoring) command. Output fields are listed in the approximate order in which they appear.

Table 116: show interfaces Output Fields (Flow Monitoring)

Field Name	Field Description	Level of Output
Physical Interface		
Physical interface	Name of the physical interface.	All levels

Table 116: show interfaces Output Fields (Flow Monitoring) ***(continued)***

Field Name	Field Description	Level of Output
Link	Status of the link: **up** or **down**.	All levels
Enabled	State of the interface. Possible values are described in the "Enabled Field" section under *Common Output Fields Description*.	All levels
Interface index	Physical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	SNMP index number for the physical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Description	Description and name of the interface.	All levels
Type	Type of interface.	All levels
Link-level type	Encapsulation type used on the physical interface.	All levels
MTU	Maximum Transmit Unit (MTU). Size of the largest packet to be transmitted.	All levels
Clocking	Reference clock source of the interface.	All levels
Speed	Network speed on the interface.	All levels
Device flags	Information about the physical device. Possible values are described in the "Device Flags" section under *Common Output Fields Description*.	All levels
Interface flags	Information about the interface. Possible values are described in the "Interface Flags" section under *Common Output Fields Description*.	All levels
Link type	Data transmission type.	All levels
Link flags	Information about the link. Possible values are described in the "Link Flags" section under *Common Output Fields Description*.	All levels
Physical info	Information about the physical interface.	All levels
Hold-times	Current interface hold-time up and hold-time down. Value is in milliseconds.	**detail extensive**
Current address	Configured MAC address.	**detail extensive** none

Table 116: show interfaces Output Fields (Flow Monitoring) ***(continued)***

Field Name	Field Description	Level of Output
Hardware address	Media access control (MAC) address of the interface.	**detail extensive** none
Alternate link address	Backup link address.	**detail extensive** none
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped: *year-month-day hour:minute:second timezone* (*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago)**	**detail extensive**
Statistics last cleared	Time when the statistics for the interface were last set to zero.	**detail extensive**
Traffic statistics	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**, **Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets**, **Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Input errors	• **Errors**—Input errors on the interface. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Frames received smaller than the runt threshold. • **Giants**—Frames received larger than the giant threshold. • **Policed Discards**—Frames that the incoming packet match code discarded because the frames did not recognize them or were not of interest. Usually, this field reports protocols that Junos does not support. • **Resource errors**—Sum of transmit drops.	**extensive**

Table 116: show interfaces Output Fields (Flow Monitoring) *(continued)*

Field Name	Field Description	Level of Output
Output errors	• **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and then up, or another problem occurs. If the number of carrier transitions increments quickly, possibly once every 10 seconds, the cable, the remote system, or the interface is malfunctioning. • **Errors**—Sum of outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet dropped by the ASIC Red mechanism. • **Resource errors**—Sum of transmit drops.	**extensive**
Logical Interface		
Logical interface	Name of the logical interface.	All levels
Index	Logical interface index number, which reflects its initialization sequence.	**detail extensive** none
SNMP ifIndex	Logical interface SNMP interface index number.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Flags	Information about the logical interface; values are described in the "Logical Interface Flags" section under *Common Output Fields Description*.	All levels
Encapsulation	Encapsulation on the logical interface.	All levels
Traffic statistics	Total number of bytes and packets received and transmitted on the logical interface. These statistics are the sum of the local and transit statistics. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize. • **Input bytes, Output bytes**—Number of bytes received and transmitted on the interface. • **Input packets, Output packets**—Number of packets received and transmitted on the interface.	**detail extensive**
Local statistics	Statistics for traffic received from and transmitted to the Routing Engine. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**

Table 116: show interfaces Output Fields (Flow Monitoring) ***(continued)***

Field Name	Field Description	Level of Output
Transit statistics	Statistics for traffic transiting the router. When a burst of traffic is received, the value in the output packet rate field might briefly exceed the peak cell rate. It takes awhile (generally, less than 1 second) for this counter to stabilize.	**detail extensive**
Protocol	Protocol family configured on the logical interface (such as **iso** or **inet6**).	**detail extensive** none
MTU	MTU size on the logical interface.	**detail extensive** none
Generation	Unique number for use by Juniper Networks technical support only.	**detail extensive**
Route table	Route table in which this address exists; for example, **Route table:0** refers to inet.0.	**detail extensive**
Flags	Information about the protocol family flags. Possible values are described in the "Family Flags" section under *Common Output Fields Description*.	**detail extensive** none

Sample Output

show interfaces extensive (Flow Monitoring)

```
user@host> show interfaces mo-4/0/0  extensive
```

```
Physical interface: mo-4/0/0, Enabled, Physical link is Up
  Interface index: 144, SNMP ifIndex: 42, Generation: 28
  Description: monitor pic 2
  Type: Adaptive-Services, Link-level type: Adaptive-Services, MTU: Unlimited,
  Clocking: Unspecified, Speed: 800mbps
  Device flags   : Present Running
  Interface flags: Point-To-Point SNMP-Traps 16384
  Link type      : Full-Duplex
  Link flags     : None
  Physical info  : Unspecified
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: Unspecified, Hardware address: Unspecified
  Alternate link address: Unspecified
  Last flapped   : 2005-05-24 16:43:12 PDT (00:17:46 ago)
  Statistics last cleared: Never
  Traffic statistics:
```

```
   Input  bytes  :            756824218              8328536 bps
   Output bytes  :            872916185              8400160 bps
   Input  packets:               508452                  697 pps
   Output packets:             15577196                18750 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Giants: 0,
    Policed discards: 0, Resource errors: 0
  Output errors:
    Carrier transitions: 2, Errors: 0, Drops: 0, MTU errors: 0,
    Resource errors: 0

  Logical interface mo-4/0/0.0 (Index 83) (SNMP ifIndex 43) (Generation 26)
    Flags: Point-To-Point SNMP-Traps Encapsulation: Adaptive-Services
    Traffic statistics:
     Input  bytes  :            756781796
     Output bytes  :            872255328
     Input  packets:               507233
     Output packets:             15575988
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
     Input  bytes  :            756781796              8328536 bps
     Output bytes  :            872255328              8400160 bps
     Input  packets:               507233                  697 pps
     Output packets:             15575988                18750 pps
    Protocol inet, MTU: Unlimited, Generation: 38, Route table: 0
      Flags: None

  Logical interface mo-4/0/0.16383 (Index 84) (SNMP ifIndex 58) (Generation 27)
...
```

show passive-monitoring error

Syntax

```
show passive-monitoring error (* | all | mo-fpc/pic/port)
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display passive monitoring error statistics.

Options

*** | all | mo-*fpc/pic/port***—Display error statistics for monitoring interfaces. Use a wildcard character, specify all interfaces, or provide a specific interface name.

Required Privilege Level

view

List of Sample Output

show passive-monitoring error all on page 1275

Output Fields

Table 117 on page 1274 lists the output fields for the **show passive-monitoring error** command. Output fields are listed in the approximate order in which they appear.

Table 117: show passive-monitoring error Output Fields

Field Name	Field Description
Passive monitoring interface	Name of the passive monitoring interface.
Local interface index	Index counter of the local interface.
Interface state	State of the passive monitoring interface: • **Monitoring**—Specified interface is actively monitoring. • **Disabled**—Specified interface has been disabled from the CLI. • **Not monitoring**—The interface is operational, but not monitoring. This condition occurs when an interface first comes online, or when the interface is operational, but no logical unit has been configured under the physical interface. • **Unknown**—Unknown state. • **Error**—An error occurred during the process of determining the state of the interface.

Table 117: show passive-monitoring error Output Fields *(continued)*

Field Name	Field Description
Error information	
Packets dropped (no memory)	Number of packets dropped because of memory shortage.
Packets dropped (not IP)	Number of non-IP packets dropped.
Packets dropped (not IPv4)	Number of packets dropped because they failed the IPv4 version check.
Packets dropped (header too small)	Number of packets dropped because the packet length or IP header length was too small.
Memory allocation failures	Number of flow record memory allocation failures. A small number reflects failures to replenish the free list. A large number indicates the monitoring station is almost out of memory space.
Memory free failures	Number of flow record memory free failures.
Memory free list failures	Number of flow records received from free list that failed. Memory is nearly exhausted or too many new flows greater than 128 KB are being created per second.
Memory warning	Whether the flows have exceeded 1 million packets per second (Mpps) on a Monitoring Services PIC or 2 Mpps on a Monitoring Services II PIC. The response can be **Yes** or **No**.
Memory overload	Whether the memory has been overloaded. The response can be **Yes** or **No.**
PPS overload	Whether the PIC is receiving more packets per second than the configured threshold. The response can be **Yes** or **No**.
BPS overload	Whether the PIC is receiving more bits per second than the configured threshold. The response can be **Yes** or **No**.

Sample Output

show passive-monitoring error all

```
user@host> show passive-monitoring error all
```

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
Interface state: Monitoring
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory warning: No, Memory overload: No, PPS overload: No, BPS overload: No

Passive monitoring interface: mo-4/1/0, Local interface index: 45
Interface state: Not monitoring
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory warning: No, Memory overload: No, PPS overload: No, BPS overload: No
```

show passive-monitoring flow

Syntax

```
show passive-monitoring flow (* | all | mo-fpc/pic/port)
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display passive flow statistics.

Options

*** | all | mo-*fpc/pic/port***—Display passive flow statistics for monitoring interfaces. Use a wildcard character, specify all interfaces, or provide a specific interface name.

Required Privilege Level

view

List of Sample Output

show passive-monitoring flow all on page 1278

Output Fields

Table 118 on page 1277 lists the output fields for the **show passive-monitoring flow** command. Output fields are listed in the approximate order in which they appear.

Table 118: show passive-monitoring flow Output Fields

Field Name	Field Description
Passive monitoring interface	Name of the passive monitoring interface.
Local interface index	Index counter of the local interface.
Interface state	State of the passive monitoring interface: • **Monitoring**—Specified interface is actively monitoring. • **Disabled**—Specified interface has been disabled from the CLI. • **Not monitoring**—The interface is operational, but not monitoring. This condition occurs when an interface first comes online, or when the interface is operational, but no logical unit has been configured under the physical interface. • **Unknown**—Unknown state. • **Error**—An error occurred during the process of determining the state of the interface.

Table 118: show passive-monitoring flow Output Fields *(continued)*

Field Name	Field Description
Flow information	
Flow packets	Number of packets received by an operational PIC.
Flow bytes	Number of bytes received by an operational PIC.
Flow packets 10-second rate	Number of packets per second handled by the PIC and displayed as a 10-second average.
Flow bytes 10-second rate	Number of bytes per second handled by the PIC and displayed as a 10-second average.
Active flows	Number of currently active flows tracked by the PIC.
Total flows	Total number of flows received by an operational PIC.
Flows exported	Total number of flows exported by an operational PIC.
Flows packets exported	Total number of cflowd packets exported by an operational PIC.
Flows inactive timed out	Total number of flows that are exported because of inactivity.
Flows active timed out	Total number of long-lived flows that are exported because of an active timeout.

Sample Output

show passive-monitoring flow all

```
user@host> show passive-monitoring flow all
```

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
Interface state: Monitoring
  Flow information
    Flow packets: 6533434, Flow bytes: 653343400
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 1599
```

```
    Flows exported: 1599, Flows packets exported: 55
    Flows inactive timed out: 1599, Flows active timed out: 0

Passive monitoring interface: mo-4/1/0, Local interface index: 45
Interface state: Monitoring
  Flow information
    Flow packets: 6537780, Flow bytes: 653778000
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 1601
    Flows exported: 1601, Flows packets exported: 55
    Flows inactive timed out: 1601, Flows active timed out: 0
```

show passive-monitoring memory

Syntax

```
show passive-monitoring memory (* | all | mo-fpc/pic/port)
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display passive monitoring memory and flow record statistics

Options

*** | all | mo-*fpc/pic/port***—Display memory and flow record statistics for monitoring interfaces. Use a wildcard character, specify all interfaces, or provide a specific interface name.

Required Privilege Level

view

List of Sample Output

show passive-monitoring memory all on page 1281

Output Fields

Table 119 on page 1280 lists the output fields for the **show passive-monitoring memory** command. Output fields are listed in the approximate order in which they appear.

Table 119: show passive-monitoring memory Output Fields

Field Name	Field Description
Passive monitoring interface	Name of the passive monitoring interface.
Local interface index	Index counter of the local interface.
Memory utilization	
Allocation count	Number of flow records allocated.
Free count	Number of flow records freed.
Maximum allocated	Maximum number of flow records allocated since the monitoring station booted. This number represents the peak number of flow records allocated at a time.

Table 119: show passive-monitoring memory Output Fields *(continued)*

Field Name	Field Description
Allocations per second	Flow records allocated per second during the last statistics interval on the PIC.
Frees per second	Flow records freed per second during the last statistics interval on the PIC.
Total memory used, Total memory free	Total memory currently used and total amount of memory currently free (in bytes).

Sample Output

show passive-monitoring memory all

`user@host>` **show passive-monitoring memory all**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  Memory utilization
    Allocation count: 1600, Free count: 1599, Maximum allocated: 1600
    Allocations per second: 3200, Frees per second: 1438
    Total memory used (in bytes): 103579176, Total memory free (in bytes):
    163914184
```

show passive-monitoring status

Syntax

```
show passive-monitoring status (*| all | mo-fpc/pic/port)
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display passive monitoring status.

Options

*** | all | mo-*fpc/pic/port***—Display status for monitoring interfaces. Use a wildcard character, specify all interfaces, or provide a specific interface name.

Required Privilege Level

view

List of Sample Output

show passive-monitoring status all on page 1283

Output Fields

Table 120 on page 1282 lists the output fields for the **show passive-monitoring status** command. Output fields are listed in the approximate order in which they appear.

Table 120: show passive-monitoring status Output Fields

Output Field	Output Field Description
Passive monitoring interface	Name of the passive monitoring interface.
Local interface index	Index counter of the local interface.
Interface state	Monitoring state of the passive monitoring interface. • **Monitoring**—PIC is actively monitoring. • **Disabled**—PIC has been disabled using the CLI. • **Not monitoring**—PIC is operational, but not monitoring. This condition can happen while the PIC is coming online, or when the PIC is operational but has no logical unit configured under the physical interface. • **Unknown**

Table 120: show passive-monitoring status Output Fields *(continued)*

Output Field	Output Field Description
Group index	Integer that represents the monitoring group of which the PIC is a member. **Group index** is a mapping from the group name to an index. It is not related to the number of monitoring groups.
Export interval	Configured export interval for cflowd records, in seconds.
Export format	Configured export format (only cflowd version 5 is supported).
Protocol	Protocol the PIC is configured to monitor (only IPv4 is supported).
Engine type	Configured engine type that is inserted in output cflowd packets.
Engine ID	Configured engine ID that is inserted in output cflowd packets.

Sample Output

show passive-monitoring status all

`user@host>` **show passive-monitoring status all**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
Interface state: Monitoring
  Group index: 0
  Export interval: 15 secs, Export format: cflowd v5
  Protocol: IPv4, Engine type: 1, Engine ID: 1

Passive monitoring interface: mo-4/1/0, Local interface index: 45
Interface state: Disabled

Passive monitoring interface: mo-4/2/0, Local interface index: 46
Interface state: Not monitoring
```

show passive-monitoring usage

Syntax

```
show passive-monitoring usage (* | all | mo-fpc/pic/port)
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display passive monitoring usage statistics.

Options

*** | all | mo-*fpc/pic/port***—Display usage statistics for monitoring interfaces. Use a wildcard character, specify all interfaces, or provide a specific interface name.

Required Privilege Level

view

List of Sample Output

show passive-monitoring usage all on page 1285

Output Fields

Table 121 on page 1284 lists the output fields for the **show passive-monitoring usage** command. Output fields are listed in the approximate order in which they appear.

Table 121: show passive-monitoring usage Output Fields

Output Field	Output Field Description
Passive monitoring interface	Name of the passive monitoring interface.
Local interface index	Index counter of the local interface.
CPU utilization	
Uptime	Time, in milliseconds, that the PIC has been operational.
Interrupt time	Total time that the PIC has spent processing packets since the last PIC reset.
Load (5 second)	CPU load on the PIC, averaged more than 5 seconds. The number is a percentage obtained by dividing the time spent on active tasks by the total elapsed time.

Table 121: show passive-monitoring usage Output Fields *(continued)*

Output Field	Output Field Description
Load (1 minute)	CPU load on the PIC, averaged more than 1 minute. The number is a percentage obtained by dividing the time spent on active tasks by the total elapsed time.

Sample Output

show passive-monitoring usage all

`user@host>` **show passive-monitoring usage**

```
Passive monitoring interface: mo-4/0/0, Local interface index: 44
  CPU utilization
    Uptime: 653155 milliseconds, Interrupt time: 40213754 microseconds
    Load (5 second): 20%, Load (1 minute): 17%

Passive monitoring interface: mo-4/1/0, Local interface index: 45
  CPU utilization
    Uptime: 652292 milliseconds, Interrupt time: 40223178 microseconds
    Load (5 second): 22%, Load (1 minute): 15%

Passive monitoring interface: mo-4/2/0, Local interface index: 46
  CPU utilization
    Uptime: 649491 milliseconds, Interrupt time: 40173645 microseconds
    Load (5 second): 22%, Load (1 minute): 10098862%
```

show services accounting aggregation

Syntax

```
show services accounting aggregation aggregation-type <aggregation-value>
<detail | extensive | terse>
<limit limit-value>
< name service-name>
<order (bytes | packets)>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display information about the aggregated active flows being processed by the accounting service.

Options

aggregation-type <aggregation-value>—Display information for the specified aggregation type and optional value:

- **as <*source-as-value* | *destination-as-value* | *input-snmp-interface-index-value* | *output-snmp-interface-index-value*>**—Aggregate by autonomous system (AS).
- **destination-prefix <*destination-prefix-value* | *destination-as-value* | *output-snmp-interface-index-value*>**—Aggregate by destination prefix.
- **protocol-port <*protocol-value* | *source-port-value* | *destination-port-value*>**—Aggregate by protocol and port.
- **source-destination-prefix <*source-prefix-value* | *destination-prefix-value* | *destination-as-value* | *source-as-value* | *input-snmp-interface-index-value* | *output-snmp-interface-index-value*>**—Aggregate by source and destination prefix.
- **source-prefix <*source-prefix-value* | *source-as-value* | *input-snmp-interface-index-value*>**—Aggregate by source prefix.

detail | extensive | terse—(Optional) Display the specified level of output.

limit *limit-value*—(Optional) Limit the display output to the specified number of flows. The default is no limit.

name *service-name*—(Optional) Display information about the aggregated flows for a specified service name.

order (bytes | packets)—(Optional) Display the flow with the ordering of the highest number, either by byte count or by packet count.

Additional Information

For information about aggregation configuration options, see the *Junos OS Services Interfaces Library for Routing Devices*.

Required Privilege Level

view

List of Sample Output

show services accounting aggregation protocol-port detail on page 1288
show services accounting aggregation source-destination-prefix on page 1289
show services accounting aggregation source-destination- prefix order packet detail on page 1289
show services accounting aggregation source-destination- prefix extensive limit on page 1290
show services accounting aggregation source-destination-prefix name terse on page 1290

Output Fields

Table 122 on page 1287 lists the output fields for the **show services accounting aggregation** command. Output fields are listed in the approximate order in which they appear.

Table 122: show services accounting aggregation Output Fields

Field Name	Field Description
Service Accounting interface	Name of the service accounting interface.
Local interface index	Index corresponding to the service accounting interface.
Service name	Name of a service that was configured at the [**edit forwarding-options accounting**] hierarchy level. The default display, **(default sampling)**, indicates the service was configured at the [**edit forwarding-options sampling-level**] hierarchy level.
Protocol	Protocol identifier and number.
Source Port	Source port identifier and number.
Destination Port	Destination port identifier and number.
Source-AS	Source autonomous system (AS) number.
Destination-AS	Destination AS number.
Source Prefix	Source prefix.

Table 122: show services accounting aggregation Output Fields *(continued)*

Field Name	Field Description
Destination Prefix	Destination prefix.
Source address	Source address.
Source prefix length	Source prefix length.
Destination address	Destination address.
Destination prefix length	Destination prefix length.
Input SNMP interface index	SNMP index of the interface the packet came in on.
Output SNMP interface index	SNMP index of the interface the packet went out on.
Start time	Actual time when the packet in this aggregation was first seen.
End time	Actual time when the packet in this aggregation was last seen.
Flow count	Number of flows in the aggregation.
Packet count	Number of packets in the aggregation.
Byte count	Number of bytes in the aggregation.

Sample Output

show services accounting aggregation protocol-port detail

`user@host>` **show service accounting aggregation protocol-port detail**

```
Service Accounting interface: mo-2/0/0, Local interface index: 468
Service name: (default sampling)
  Protocol: 6, Source port: 20, Destination port: 20
```

```
  Start time: 442349, End time: 6425714
  Flow count: 194, Packet count: 4294964388, Byte count: 4294781184

  Protocol: 0, Source port: 0, Destination port: 0
  Start time: 442349, End time: 6425749
  Flow count: 204, Packet count: 4294964324, Byte count: 4294777088

  Protocol: 17, Source port: 123, Destination port: 123
  Start time: 442364, End time: 6425784
  Flow count: 186, Packet count: 4294964152, Byte count: 4294766080
```

show services accounting aggregation source-destination-prefix

`user@host>` **show service accounting aggregation source-destination-prefix**

```
Service Accounting interface: rsp0, Local interface index: 171
Service name: (default sampling)
Interface state: Accounting
Source         Destination       Input        Output      Flow   Packet  Byte
prefix         prefix            interface    interface   count  count   count
192.0.2.0/20   198.51.100.0/24   ge-5/0/1.0   ge-5/0/0.0  256    491761  31472704
192.0.2.0/20   203.0.113.36/32   ge-5/0/1.0   ge-5/0/0.0  1      1926    123264
192.0.2.0/20   203.0.113.59/32   ge-5/0/1.0   ge-5/0/0.0  1      1926    123264
192.0.2.0/20   192.168.0.63/32   ge-5/0/1.0   ge-5/0/0.0  1      1925    123200
192.0.2.0/20   192.168.0.32/32   ge-5/0/1.0   ge-5/0/0.0  1      1925
```

show services accounting aggregation source-destination- prefix order packet detail

`user@host>` **show service accounting aggregation source-destination-prefix order packet detail name t2 input-snmp-interface-index 538**

```
Service Accounting interface: mo-2/0/0, Local interface index: 468
Service name: t2
Source         Destination  Input SNMP  Output  SNMP   Flow   Packet Byte
Prefix            Prefix       Index      Index Count  Count    Count
10.1.1.2/20   192.168.167.1/0  538         432    1     60    46483
10.1.1.2/20   192.168.168.1/0  538         432    1     60     5191
10.1.1.2/20   192.168.154.1/0  538         432    2     60    45504
10.1.1.2/20   192.168.76.1/0   538         432    1     60    42177
10.1.1.2/20   192.168.149.1/0  538         432    1     60    49184
10.1.1.2/20   192.168.113.1/0  538         432    2     60    48757
```

show services accounting aggregation source-destination- prefix extensive limit

`user@host>` **show service accounting aggregation source-destination-prefix name t2 extensive limit 3**

```
Service Accounting interface: mo-2/0/0, Local interface index: 542
Service name: t2

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 192.168.200.176.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 5340

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 192.168.160.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 5490

  Source address: 10.1.1.2, Source prefix length: 20
  Destination address: 192.168.160.1, Destination prefix length: 0
  Input SNMP interface index: 24, Output SNMP interface index: 26
  Source-AS: 69, Destination-AS: 69
  Start time: Fri Feb 21 14:16:57 2003, End time: Fri Feb 21 14:22:50 2003
  Flow count: 0, Packet count: 6, Byte count: 4079
```

show services accounting aggregation source-destination-prefix name terse

`user@host>` **show service accounting aggregation source-destination-prefix name T3 terse**

```
Service Accounting interface: rsp0, Local interface index: 171
Service name: T3
Interface state: Accounting
Source        Destination       Input        Output       Flow   Packet  Byte
prefix        prefix            interface    interface    count  count   count
10.1.0.0/20   192.168.3.0/24    ge-5/0/1.0   ge-5/0/0.0   256    639822  40948608
10.1.0.0/20   192.168.2.67/32   ge-5/0/1.0   ge-5/0/0.0   1      2485    159040
10.1.0.0/20   192.168.2.92/32   ge-5/0/1.0   ge-5/0/0.0   1      2485
```

show services accounting aggregation template

Syntax

```
show services accounting aggregation template
<template-name template-name>
```

Release Information

Command introduced in Junos OS Release 8.3.

Description

Display information for flow aggregation version 9 templates.

Options

none—Display information for all flow aggregation version version 9 templates.

template-name *template-name*—(Optional) Display information for the specified template only.

Required Privilege Level

view

List of Sample Output

show services accounting aggregation template template-name on page 1292

Output Fields

Table 123 on page 1291 lists the output fields for the **show services accounting aggregation template** command. Output fields are listed in the approximate order in which they appear.

Table 123: show services accounting aggregation template Output Fields

Field Name	Field Description
MPLS Label 1	Position of first MPLS label.
MPLS Label 2	Position of second MPLS label.
MPLS Label 3	Position of third MPLS label.
MPLS Top Level Address	Outer top label FEC IP address.
Packet Count	Number of packets sent.

Sample Output

show services accounting aggregation template template-name

user@host> **show services accounting aggregation template template-name mpls**

```
MPLS label 1: 299808, MPLS label 2: 0, MPLS label 3: 0
Source address: 192.0.2.2, Destination address: 10.255.15.22, Top Label Address:
198.51.100.10
Source port: 0, Destination port: 0
Protocol: 61, TOS: 0, TCP flags: 0
Source mask: 24, Destination mask: 32
Input SNMP interface index: 503, Output SNMP interface index: 505
Start time: 40780, End time: 157330
Packet count: 3949198, Byte count: 181663062
```

show services accounting errors

Syntax

```
show services accounting errors
<inline-jflow | name (* | all | service-name)>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display active flow error statistics.

Options

none—Display error statistics for all services accounting instances.

inline-jflow fpc-slot *slot-number*—(Optional) Display error statistics for inline jflow.

name (* | all | *service-name*)—(Optional) Display active flow error statistics. Use a wildcard character, specify all services, or provide a specific service name.

Required Privilege Level

view

RELATED DOCUMENTATION

show services accounting flow | 1299

List of Sample Output

Output Fields

Table 124 on page 1294 lists the output fields for the **show services accounting errors** command. Output fields are listed in the approximate order in which they appear.

Table 124: show services accounting errors Output Fields

Field	Field Description
Service Accounting interface	Name of the service accounting interface.
Local interface index	Index counter of the local interface.
FPC slot	Slot number of the FPC for which the flow information is displayed. (Available only when the **inline-jflow fpc-slot *slot-number*** option is used.)
Service name	Name of a service that was configured at the [**edit forwarding-options accounting**] hierarchy level. The default display, **(default sampling)**, indicates the service was configured at the [**edit forwarding-options sampling-level**] hierarchy level.
Error Information	
Packets dropped (no memory)	Number of packets dropped because of memory shortage.
Packets dropped (not IP)	Number of non-IP packets dropped.
Packets dropped (not IPv4)	Number of packets dropped because they failed the IPv4 version check.
Packets dropped (header too small)	Number of packets dropped because the packet length or IP header length was too small.
Memory allocation failures	Number of flow record memory allocation failures. A small number reflects failures to replenish the free list. A large number indicates the monitoring station is almost out of memory space.
Memory free failures	Number of flow record memory free failures.
Memory free list failures	Number of flow records received from the free list that failed. Memory is nearly exhausted, or too many new flows greater than 128 KB are being created per second.
Memory overload	Whether the memory has been overloaded. The response can be **Yes** or **No**.
PPS overload	Whether the PIC is receiving more packets per second than the configured threshold. The response can be **Yes** or **No**.

Table 124: show services accounting errors Output Fields *(continued)*

Field	Field Description
BPS overload	Whether the PIC is receiving more bits per second than the configured threshold. The response can be **Yes** or **No**.
Flow Creation Failures	Number of times flow creation failed.
Route Record Lookup Failures	Number of times the route record lookup failed.
AS Lookup Failures	Number of times autonomous system lookup failed.
Export Packet Failures	Number of times packet export failed.

Sample Output

show services accounting errors (Monitoring PIC interface)

`user@host>` **show services accounting errors**

```
Service Accounting interface: mo-1/1/0, Local interface index: 15
Service name: (default sampling)
  Error information
    Packets dropped (no memory): 0, Packets dropped (not IP): 0
    Packets dropped (not IPv4): 0, Packets dropped (header too small): 0
    Memory allocation failures: 0, Memory free failures: 0
    Memory free list failures: 0
    Memory overload: No, PPS overload: No, BPS overload: No
```

Sample Output

show services accounting errors (Service PIC interface)

`user@host>` **show services accounting errors**

```
Service Accounting interface: sp-0/1/0
Service name: (default sampling)
  Error information
    Service sets dropped: 0, Active timeout failures: 0
    Export packet failures: 0, Flow creation failures: 0
    Memory overload: No
```

```
Service Accounting interface: sp-1/0/0
Service name: (default sampling)
  Error information
    Service sets dropped: 0, Active timeout failures: 0
    Export packet failures: 0, Flow creation failures: 0
    Memory overload: No
```

show services accounting errors inline-jflow fpc-slot (When Only IPv6 Is Configured)

```
user@host> show services accounting errors inline-jflow fpc-slot 5
```

```
  Error information
    FPC Slot: 5
    Flow Creation Failures: 0
    Route Record Lookup Failures: 0, AS Lookup Failures: 0
    Export Packet Failures: 0
    Memory Overload: No, Memory Alloc Fail Count: 0
```

show services accounting errors inline-jflow fpc-slot (When IPv4, IPv6, VPLS, and Bridge Are Configured)

```
user@host> show services accounting errors inline-jflow fpc-slot 5
```

```
  Error information
    FPC Slot: 5
    Flow Creation Failures: 0
    Route Record Lookup Failures: 0, AS Lookup Failures: 0
    Export Packet Failures: 0
    Memory Overload: No, Memory Alloc Fail Count: 0

    IPv4:
    IPv4 Flow Creation Failures: 0
    IPv4 Route Record Lookup Failures: 0, IPv4 AS Lookup Failures: 0
    IPv4 Export Packet Failures: 0

    IPv6:
```

```
    IPv6 Flow Creation Failures: 0
    IPv6 Route Record Lookup Failures: 0, IPv6 AS Lookup Failures: 0
    IPv6 Export Packet Failures: 0


    VPLS:
    VPLS Flow Creation Failures: 0
    VPLS Export Packet Failures: 0

    BRIDGE:
    BRIDGE Flow Creation Failures: 0
    BRIDGE Route Record Lookup Failures: 0, BRIDGE AS Lookup Failures: 0
    BRIDGE Export Packet Failures: 0
```

show services accounting errors inline-jflow (MX80 Router When Both IPv4 and IPv6 Are Configured)

```
user@host> show services accounting errors inline-jflow
```

```
  Error information
    TFEB Slot: 0
    Flow Creation Failures: 0
    Route Record Lookup Failures: 0, AS Lookup Failures: 0
    Export Packet Failures: 0
    Memory Overload: No

   IPv4:
    IPv4 Flow Creation Failures: 0
    IPv4 Route Record Lookup Failures: 0, IPv4 AS Lookup Failures: 0
    IPv4 Export Packet Failures: 0

    IPv6:
    IPv6 Flow Creation Failures: 0
    IPv6 Route Record Lookup Failures: 0, IPv6 AS Lookup Failures: 0
    IPv6 Export Packet Failures: 0
```

show services accounting errors inline-jflow fpc-slot(PTX1000 Router When Both IPv4 and IPv6 Are Configured)

```
user@host> show services accounting errors inline-jflow fpc-slot 0
```

```
Error information
FPC Slot: 0
Flow Creation Failures: 0
```

```
Route Record Lookup Failures: 0, AS Lookup Failures: 0
Export Packet Failures: 0
Memory Overload: No, Memory Alloc Fail Count: 0

IPv4:
IPv4 Flow Creation Failures: 0
IPv4 Route Record Lookup Failures: 0, IPv4 AS Lookup Failures: 0
IPv4 Export Packet Failures: 0

IPv6:
IPv6 Flow Creation Failures: 0
IPv6 Route Record Lookup Failures: 0, IPv6 AS Lookup Failures: 0
IPv6 Export Packet Failures: 0
```

show services accounting errors inline-jflow (SRX Series Devices When Both IPv4 and IPv6 Are Configured)

```
user@host> show services accounting errors inline-jflow
```

```
Error information
    FPC Slot: 0
    Flow Creation Failures: 0
    Route Record Lookup Failures: 0, AS Lookup Failures: 0
    Export Packet Failures: 0
    Memory Overload: No, Memory Alloc Fail Count: 0

    IPv4:
    IPv4 Flow Creation Failures: 0
    IPv4 Route Record Lookup Failures: 0, IPv4 AS Lookup Failures: 0
    IPv4 Export Packet Failures: 0
```

show services accounting flow

Syntax

```
show services accounting flow
<inline-jflow fpc-slot slot-number | logical-system (all | logical-system) | name (* | all | service-name)>
```

Release Information

Command introduced before Junos OS Release 7.4.

Junos OS Release 10.0 added the capability to display output from multiple sampling instances.

Description

Display active flow statistics.

Options

none—Display active flow statistics for all service instances.

logical-system (all | *logical-system*)—(Optional) Display active flow statistics for the specified logical system or all logical systems on the device.

inline-jflow (fpc-slot *slot-number*)—(Optional) Display inline flow statistics for the specified FPC.

name (* | all | *service-name*)—(Optional) Display services accounting active flow statistics. Use a wildcard character, specify all services, or provide a specific service name.

Required Privilege Level

view

RELATED DOCUMENTATION

show services accounting status | 1317

List of Sample Output

show services accounting flow inline-jflow fpc-slot (PTX1000 Router When Both IPv4 and IPv6 Are Configured) on page 1305
show services accounting flow inline-jflow (SRX Series When IPv4 is configured) on page 1305

Output Fields

Table 125 on page 1300 lists the output fields for the **show services accounting flow** command. Output fields are listed in the approximate order in which they appear.

Table 125: show services accounting flow Output Fields

Output Field	Output Field Description
Service Accounting interface	Name of the service accounting interface.
Local interface index	Index counter of the local interface.
Service name	Name of a service that was configured at the **[edit forwarding-options accounting]** hierarchy level. The default display, **(default sampling)**, indicates the service was configured at the [**edit forwarding-options sampling-level**] hierarchy level.
Flow Information	
FPC Slot	Slot number of the FPC for which the flow information is displayed. (Available only when the **inline-jflow fpc-slot *slot-number*** option is used.)
Flow packets	Number of packets received by an operational PIC.
Flow bytes	Number of bytes received by an operational PIC.
Flow packets 10-second rate	Number of packets per second handled by the PIC and displayed as a 10-second average.
Flow bytes 10-second rate	Number of bytes per second handled by the PIC and displayed as a 10-second average.
Active flows	Number of currently active flows tracked by the PIC.
Total flows	Total number of flows received by an operational PIC.
Flows exported	Total number of flows exported by an operational PIC.
Flows packets exported	Total number of cflowd packets exported by an operational PIC.

Table 125: show services accounting flow Output Fields *(continued)*

Output Field	Output Field Description
Flows inactive timed out	Total number of flows that are exported because of inactivity.
Flows active timed out	Total number of long-lived flows that are exported because of an active timeout.

Sample Output

show services accounting flow (Flow Aggregation v5/v8 Configuration)

```
user@host> show services accounting flow
```

```
Service Accounting interface: rsp0, Local interface index: 171
Service name: (default sampling)
Interface state: Accounting
  Flow information
    Flow packets: 87168293, Flow bytes: 5578770752
    Flow packets 10-second rate: 45762, Flow bytes 10-second rate: 2928962
    Active flows: 1000, Total flows: 2000
    Flows exported: 19960, Flows packets exported: 582
    Flows inactive timed out: 1000, Flows active timed out: 29000
```

show services accounting flow (Flow Aggregation v9 Configuration)

```
user@host> show services accounting flow
```

```
  Flow information
    Service Accounting interface: sp-7/1/0, Local interface index: 149
    Flow packets: 0, Flow bytes: 0
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 0
    Flows exported: 0, Flows packets exported: 1
    Flows inactive timed out: 0, Flows active timed out: 0
```

show services accounting flow name

```
user@host> show services accounting flow name count2
```

```
Service Accounting interface: mo-1/1/0, Local interface index: 15
Service name: count2
  Flow information
    Flow packets: 0, Flow bytes: 0
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 0
    Flows exported: 0, Flows packets exported: 0
    Flows inactive timed out: 0, Flows active timed out: 0
```

show services accounting flow name all

user@host> **show services accounting flow name all**

```
Service Accounting interface: rsp0, Local interface index: 171
Service name: T2
Interface state: Accounting
  Flow information
    Flow packets: 37609891, Flow bytes: 2407033024
    Flow packets 10-second rate: 45762, Flow bytes 10-second rate: 2928953
    Active flows: 1000, Total flows: 1000
    Flows exported: 6705, Flows packets exported: 198
    Flows inactive timed out: 0, Flows active timed out: 13000

Service Accounting interface: rsp0, Local interface index: 171
Service name: T3
Interface state: Accounting
  Flow information
    Flow packets: 37750807, Flow bytes: 2416051712
    Flow packets 10-second rate: 45762, Flow bytes 10-second rate: 2928940
    Active flows: 1000, Total flows: 1000
    Flows exported: 13437, Flows packets exported: 378
    Flows inactive timed out: 0, Flows active timed out: 13000

Service Accounting interface: rsp0, Local interface index: 171
Service name: T4
 Interface state: Accounting
  Flow information
    Flow packets: 0, Flow bytes: 0
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 0
    Flows exported: 0, Flows packets exported: 0
    Flows inactive timed out: 0, Flows active timed out: 0

Service Accounting interface: rsp0, Local interface index: 171
```

```
Service name: count1
Interface state: Accounting
  Flow information
    Flow packets: 0, Flow bytes: 0
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 0
    Flows exported: 0, Flows packets exported: 0
    Flows inactive timed out: 0, Flows active timed out: 0
```

show services accounting flow (Multiple Sampling Instances)

`user@host>` **show services accounting flow**

```
  Flow information
    Service Accounting interface: sp-2/0/0, Local interface index: 215
    Flow packets: 9867, Flow bytes: 631488
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 628
    Active flows: 2, Total flows: 10
    Flows exported: 4028, Flows packets exported: 6150
    Flows inactive timed out: 8, Flows active timed out: 4026

    Service Accounting interface: sp-2/1/0, Local interface index: 223
    Flow packets: 0, Flow bytes: 0
    Flow packets 10-second rate: 0, Flow bytes 10-second rate: 0
    Active flows: 0, Total flows: 0
    Flows exported: 0, Flows packets exported: 1
    Flows inactive timed out: 0, Flows active timed out: 0
```

show services accounting flow inline-jflow fpc-slot (for IPv4 Flow)

`user@host>` **show services accounting flow inline-jflow fpc-slot 5**

```
  Flow information
    FPC Slot: 5
    Flow Packets: 0, Flow Bytes: 0
    Active Flows: 0, Total Flows: 0
    Flows Exported: 0, Flow Packets Exported: 0
    Flows Inactive Timed Out: 0, Flows Active Timed Out: 0
```

show services accounting flow inline-jflow fpc-slot (with IPv4, IPv6, VPLS, and Bridge Configuration)

`user@host>` **show services accounting flow inline-jflow fpc-slot 5**

```
Flow information
    FPC Slot: 5
    Flow Packets: 0, Flow Bytes: 0
    Active Flows: 0, Total Flows: 0
    Flows Exported: 0, Flow Packets Exported: 0
    Flows Inactive Timed Out: 0, Flows Active Timed Out: 0

    IPv4 Flows:
    IPv4 Flow Packets: 0, IPv4 Flow Bytes: 0
    IPv4 Active Flows: 0, IPv4 Total Flows: 0
    IPv4 Flows Exported: 0, IPv4 Flow Packets exported: 0
    IPv4 Flows Inactive Timed Out: 0, IPv4 Flows Active Timed Out: 0

    IPv6 Flows:
    IPv6 Flow Packets: 0, IPv6 Flow Bytes: 0
    IPv6 Active Flows: 0, IPv6 Total Flows: 0
    IPv6 Flows Exported: 0, IPv6 Flow Packets Exported: 0
    IPv6 Flows Inactive Timed Out: 0, IPv6 Flows Active Timed Out: 0

    VPLS Flows:
    VPLS Flow Packets: 0, VPLS Flow Bytes: 0
    VPLS Active Flows: 0, VPLS Total Flows: 0
    VPLS Flows Exported: 0, VPLS Flow Packets Exported: 0
    VPLS Flows Inactive Timed Out: 0, VPLS Flows Active Timed Out: 0

    BRIDGE Flows:
    BRIDGE Flow Packets: 0, BRIDGE Flow Bytes: 0
    BRIDGE Active Flows: 0, BRIDGE Total Flows: 0
    BRIDGE Flows Exported: 0, BRIDGE Flow Packets Exported: 0
    BRIDGE Flows Inactive Timed Out: 0, BRIDGE Flows Active Timed Out: 0
    BRIDGE Flow Insert Count: 0
```

show services accounting flow inline-jflow (MX80 Router with IPv4 and IPv6 Configuration)

`user@host>` **show services accounting flow inline-jflow**

```
Flow information
    TFEB Slot: 0
    Flow Packets: 0, Flow Bytes: 0
    Active Flows: 0, Total Flows: 0
    Flows Exported: 0, Flow Packets Exported: 0
    Flows Inactive Timed Out: 0, Flows Active Timed Out: 0

    IPv4 Flows:
```

```
    IPv4 Flow Packets: 0, IPv4 Flow Bytes: 0
    IPv4 Active Flows: 0, IPv4 Total Flows: 0
    IPv4 Flows Exported: 0, IPv4 Flow Packets exported: 0
    IPv4 Flows Inactive Timed Out: 0, IPv4 Flows Active Timed Out: 0

    IPv6 Flows:
    IPv6 Flow Packets: 0, IPv6 Flow Bytes: 0
    IPv6 Active Flows: 0, IPv6 Total Flows: 0
    IPv6 Flows Exported: 0, IPv6 Flow Packets Exported: 0
    IPv6 Flows Inactive Timed Out: 0, IPv6 Flows Active Timed Out: 0
```

show services accounting flow inline-jflow fpc-slot (PTX1000 Router When Both IPv4 and IPv6 Are Configured)

`user@host>` **show services accounting flow inline-jflow fpc-slot 0**

```
Flow information
FPC Slot: 0
Flow Packets: 47427946, Flow Bytes: 5217074060
Active Flows: 0, Total Flows: 2
Flows Exported: 194, Flow Packets Exported: 7045
Flows Inactive Timed Out: 2, Flows Active Timed Out: 192

IPv4 Flows:
IPv4 Flow Packets: 47427946, IPv4 Flow Bytes: 5217074060
IPv4 Active Flows: 0, IPv4 Total Flows: 2
IPv4 Flows Exported: 194, IPv4 Flow Packets exported: 7045
IPv4 Flows Inactive Timed Out: 2, IPv4 Flows Active Timed Out: 192

IPv6 Flows:
IPv6 Flow Packets: 0, IPv6 Flow Bytes: 0
IPv6 Active Flows: 0, IPv6 Total Flows: 0
IPv6 Flows Exported: 0, IPv6 Flow Packets Exported: 0
IPv6 Flows Inactive Timed Out: 0, IPv6 Flows Active Timed Out: 0
```

show services accounting flow inline-jflow (SRX Series When IPv4 is configured)

`user@host>` **show services accounting flow inline-jflow**

```
Flow information
    FPC Slot: 0
    Flow Packets: 462680, Flow Bytes: 45433206
    Active Flows: 34, Total Flows: 61093
```

```
Flows Exported: 138936, Flow Packets Exported: 96649
Flows Inactive Timed Out: 61083, Flows Active Timed Out: 138936
Total Flow Insert Count: 0

IPv4 Flows:
IPv4 Flow Packets: 462680, IPv4 Flow Bytes: 45433206
IPv4 Active Flows: 34, IPv4 Total Flows: 61093
IPv4 Flows Exported: 138936, IPv4 Flow Packets exported: 96649
IPv4 Flows Inactive Timed Out: 61083, IPv4 Flows Active Timed Out: 138936
IPv4 Flow Insert Count: 0
```

show services accounting flow-detail

Syntax

```
show services accounting flow-detail
<detail | extensive | terse>
<filters>
<limit limit-value>
<name (* | all | service-name)>
<order (bytes | packets)>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display information about the flows being processed by the accounting service.

Options

none—Display information about all flows.

detail | extensive | terse—(Optional) Display the specified level of output.

filters—(Optional) Filter the display output of the currently active flow records. The following filters query actively changing data structures and result in different results for multiple invocations:

- **destination-as**—Display flow records filtered by destination autonomous system information.
- **destination-port**—Display flow records filtered by destination port information.
- **destination-prefix**—Display flow records filtered by destination prefix information.
- **input-snmp-interface-index**—Display flow records filtered by SNMP input interface index information.
- **output-snmp-interface-index**—Display flow records filtered by SNMP output interface index information.
- **proto**—Display flow records filtered by protocol type.
- **source-as**—Display flow records filtered by source autonomous system information.
- **source-port**—Display flow records filtered by source port information.
- **source-prefix**—Display flow records filtered by source prefix information.
- **tos**—Display flow records filtered by type of service classification.

limit *limit-value*—(Optional) Limit the display output to the specified number of flows. The default is no limit.

name (* | all | *service-name*)—(Optional) Display information about the flows being processed. Use a wildcard character, specify all services, or provide a specific services name.

order (bytes | packets)—(Optional) Display the flow with the ordering of the highest number, either by byte count or by packet count.

Additional Information

When no PIC is active, or when no route record has been downloaded from the PIC, this command reports no flows, even though packets are being sampled. This command displays information about two concurrent sessions only. If a third session is attempted, the command pauses with no output until one of the previous sessions is completed.

Required Privilege Level

view

List of Sample Output

show services accounting flow-detail on page 1310
show services accounting flow-detail limit on page 1310
show services accounting flow-detail name extensive on page 1311
show services accounting flow-detail limit order bytes on page 1311
show services accounting flow-detail name detail source-port on page 1312

Output Fields

Table 126 on page 1308 lists the output fields for the **show services accounting flow-detail** command. Output fields are listed in the approximate order in which they appear.

Table 126: show services accounting flow-detail Output Fields

Field Name	Field Description	Output Level
Service Accounting interface	Name of the service accounting interface.	All levels
Service name	Name of a service that was configured at the **[edit forwarding-options accounting]** hierarchy level. The default display, **(default sampling)**, indicates the service was configured at the **[edit forwarding-options sampling]** hierarchy level.	All levels
Local interface index	Index counter of the local interface.	All levels
TOS	Type-of-service value from the IP header.	**extensive**
Input SNMP interface index	SNMP index of the interface on which the packet came in.	**extensive**

Table 126: show services accounting flow-detail Output Fields *(continued)*

Field Name	Field Description	Output Level
Output SNMP interface index	SNMP index of the interface on which the packet went out.	**extensive**
Source-AS	Source AS number.	**extensive**
Destination-AS	Destination AS number.	**extensive**
Protocol	Name of the protocol used for the packet flow from the corresponding source address.	All levels
Input interface	Interface on which the packets were received.	All levels
Output interface	Interface on which the packets were transmitted.	All levels
TCP flags	Number of TCP header flags detected in the flow.	**extensive**
Source address	Address where the flow originated.	All levels
Source port	Name of the source port.	All levels
Source prefix length	Source prefix length.	**extensive**
Destination address	Address where the flow is sent.	All levels
Destination prefix length	Destination prefix length.	**extensive**
Destination port	Name of the destination port.	All levels
Start time	Actual time when the packet in this aggregation was first seen.	**detail extensive**
End time	Actual time when the packet in this aggregation was last seen.	**detail extensive**
Packet count	Number of packets in the aggregation.	All levels
Byte count	Number of bytes in the aggregation.	All levels
Time since last active timeout	Amount of time elapsed since the last active timeout, in the format ***hh*:*mm*:*ss***.	None specified

Table 126: show services accounting flow-detail Output Fields *(continued)*

Field Name	Field Description	Output Level
Packet count for last active timeout	Number of packets in the aggregation since the last active timeout.	None specified
Byte count for last active timeout	Number of bytes in the aggregation since the last active timeout.	None specified

Sample Output

show services accounting flow-detail

In this sample, the output is split into three sections, with ellipses (...) indicating where the sections are continued.

`user@host>` **show services accounting flow-detail**

```
Service Accounting interface: rsp0, Local interface index: 171
Service name: (default sampling)
Interface state: Accounting
Protocol   Input          Source                   Source   Output
           interface      address                    port   interface...
tcp(6)     ge-5/0/1.0     192.0.2.2                     0   ge-5/0/0.0
tcp(6)     ge-5/0/1.0     192.0.2.2                     0   ge-5/0/0.0

Destination          Destination      Packet        Byte  Time since last
address                     port       count       count   active timeout...
198.51.100.149                 0        2660      170240         00:00:58
198.51.100.138                 0        2660      170240         00:00:58

Packet count for       Byte count for
last active timeout    last active timeout
2805                   179520
2805                   179520
```

show services accounting flow-detail limit

In this sample, the output is split into three sections, with ellipses (...) indicating where the sections are continued.

user@host> **show services accounting flow-detail limit 1**

```
Service Accounting interface: rsp0, Local interface index: 171
Service name: (default sampling)
Interface state: Accounting
Protocol   Input          Source                   Source   Output
           interface      address                  port     interface...
tcp(6)     ge-5/0/1.0     192.0.2.2                      0   ge-5/0/0.0

Destination          Destination      Packet        Byte    Time since last
address              port             count         count   active timeout...
 198.51.100.149                  0        2158        138112         00:00:47

Packet count for   Byte count for
last active timeout   last active timeout
         2827                180928
```

show services accounting flow-detail name extensive

user@host> **show services accounting flow-detail name cf-2 extensive**

```
Service Accounting interface: mo-0/2/0, Local interface index: 145
Service name: cf-2
  TOS: 0, Protocol: udp(17), TCP flags: 0
  Source address: 10.10.10.1, Source prefix length: 0, Destination address:
203.0.113.20,
Destination prefix length: 0, Source port: 1173, Destination port: 69
  Input SNMP interface index: 65, Output SNMP interface index: 0, Source-AS: 0,
Destination-AS: 0
  Start time: 62425, End time: 635265, Packet count: 165845, Byte count: 9453165
```

show services accounting flow-detail limit order bytes

The output of the following command is displayed over 141 columns, not the standard 80 columns. In this sample, the output is split into three sections, with ellipses (...) indicating where the sections are continued.

user@host> **show services accounting flow-detail limit 5 order bytes**

```
Service Accounting interface: mo-2/0/0, Local interface index: 356
Service name: (default sampling)
           Input          Source                   Source   Output
Protocol   interface      address                  port     interface...
icmp(1)    ge-2/3/0.0     192.0.2.2                      0 .local.
```

```
icmp(1)    ge-2/3/0.0     192.0.2.2                    0 .local.
icmp(1)    ge-2/3/0.0     192.0.2.2                    0 .local.
icmp(1)    ge-2/3/0.0     192.0.2.2                    0 .local.
icmp(1)    ge-2/3/0.0     192.0.2.2                    0 .local.

Destination          Destination       Packet       Byte    Time since last
address              port              count        count   active timeout...
192.168.128.2                 0          16        12148   Not applicable
192.168.144.2                 0          16        15229   Not applicable
192.168.192.2                 0          16        13296   Not applicable
192.168.16.2                  0          16        13924   Not applicable
192.168.48.2                  0          16        13428   Not applicable

Packet count for          Byte count for
last active timeout       last active timeout
Not applicable            Not applicable
Not applicable            Not applicable
Not applicable            Not applicable
Not applicable            Not applicable
Not applicable            Not applicable
```

show services accounting flow-detail name detail source-port

user@host> **show services accounting flow-detail name cf-2 detail source-port 1173**

```
Service Accounting interface: mo-0/2/0, Local interface index: 145
Service name: cf-2
  Protocol: udp(17), Source address: 10.10.10.1, Source port: 1173, Destination
address:
203.0.113.20, Destination port: 69
  Start time: 62425, End time: 811115, Packet count: 142438, Byte count: 8118966
```

show services accounting memory

Syntax

```
show services accounting memory
```

Release Information
Command introduced before Junos OS Release 7.4.

Description
Display memory and flow record statistics.

Options
This command has no options.

Required Privilege Level
view

List of Sample Output
show services accounting memory (Monitoring PIC Interface) on page 1314
show services accounting memory (Service PIC Interface) on page 1314

Output Fields
Table 127 on page 1313 lists the output fields for the **show services accounting memory** command. Output fields are listed in the approximate order in which they appear.

Table 127: show services accounting memory Output Fields

Output Field	Output Field Description
Service Accounting interface	Name of the service accounting interface.
Memory Utilization	
Local interface index	Index counter of the local interface.
Allocation count	Number of flow records allocated.
Free count	Number of flow records freed.
Maximum allocated	Maximum number of flow records allocated since the monitoring station booted. This number represents the peak number of flow records allocated at a time.
Allocations per second	Flow records allocated per second during the last statistics interval on the PIC.

Table 127: show services accounting memory Output Fields *(continued)*

Output Field	Output Field Description
Frees per second	Flow records freed per second during the last statistics interval on the PIC.
Total memory used	Total amount of memory currently used (in bytes).
Total memory free	Total amount of memory currently free (in bytes).

Sample Output

show services accounting memory (Monitoring PIC Interface)

```
user@host> show services accounting memory
```

```
Service Accounting interface: mo-2/0/0, Local interface index: 468
  Memory utilization
    Allocation count: 437340, Free count: 433699, Maximum allocated: 6782
    Allocations per second: 3366, Frees per second: 6412
    Total memory used (in bytes): 133460320,
    Total memory free (in bytes): 133918352
```

show services accounting memory (Service PIC Interface)

```
user@host> show services accounting memory
```

```
Service Accounting interface: sp-0/1/0
  Memory utilization
    Allocation count: 1000, Free count: 0
    Allocations per second: 0, Frees per second: 0
    Total memory used (in bytes): 218158272
    Total memory free (in bytes): 587147696
```

```
Service Accounting interface: sp-1/0/0
  Memory utilization
    Allocation count: 1000, Free count: 0
    Allocations per second: 0, Frees per second: 0
    Total memory used (in bytes): 218157592
    Total memory free (in bytes): 587148376
```

show services accounting packet-size-distribution

Syntax

```
show services accounting packet-size-distribution
<name (* | all | service-name)>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display a packet size distribution histogram.

Options

none—Display a packet size distribution histogram of all accounting services.

name (* | all | *service-name*)—(Optional) Display a packet size distribution histogram. Use a wildcard character, specify all services, or provide a specific services name.

Required Privilege Level

view

List of Sample Output

show services accounting packet-size-distribution name on page 1316

Output Fields

Table 128 on page 1315 lists the output fields for the **show services accounting packet-size-distribution** command. Output fields are listed in the approximate order in which they appear.

Table 128: show services accounting packet-size-distribution Output Fields

Field Name	Field Description
Service Accounting interface	Name of the service accounting interface.
Service name	Name of a service that was configured at the [**edit-forwarding-options accounting**] hierarchy level. The default display, **(default sampling)**, indicates the service was configured at the [**edit-forwarding-options sampling-level**] hierarchy level.
Local interface index	Index counter of the local interface.
Range start	Smallest packet length (in bytes) to count.
Range end	Largest packet length (in bytes) to count.

Table 128: show services accounting packet-size-distribution Output Fields *(continued)*

Field Name	Field Description
Number of packets	Count of packets detected in the size between **Range start** and **Range end**.
Percentage packets	Percentage of the total number of packets that are in this size range.

Sample Output

show services accounting packet-size-distribution name

`user@host>` **show services accounting packet-size-distribution name test3**

```
Service Accounting interface: mo-0/2/0, Local interface index: 163
Service name: test3
Range start     Range end     Number of packets     Percentage packets
          32            64                  2924                    100
```

show services accounting status

Syntax

```
show services accounting status
<inline-jflow fpc-slot slot-number | name (* | all | service-name)>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 13.2R2 for EX Series switches.

Description

Display available Physical Interface Cards (PICs) for accounting services.

Options

none—Display available PICs for all accounting services.

inline-jflow fpc-slot *slot-number*—(Optional) Display inline flow accounting status for the specified FPC. For a two-member MX Series Virtual Chassis or EX9200 Virtual Chassis, the master router or switch uses FPC slot numbers 0 through 11 with no offset; the backup router or switch uses FPC slot numbers 12 through 23, with an offset of 12.

name (* | all | *service-name*)—(Optional) Display available PICs. Use a wildcard character, specify all services, or provide a specific services name.

Required Privilege Level

view

RELATED DOCUMENTATION

show services accounting flow | 1299

Inline Flow Monitoring for Virtual Chassis Overview

List of Sample Output

show services accounting status inline-jflow (SRX Series Devices When Both IPv4 and IPv6 Are Configured) on page 1321

Output Fields

Table 129 on page 1318 lists the output fields for the **show services accounting status** command. Output fields are listed in the approximate order in which they appear.

Table 129: show services accounting status Output Fields

Field	Field Description
Service Accounting interface	Name of the service accounting interface.
Service name	Name of a service that was configured at the [**edit-forwarding-options accounting**] hierarchy level. The default display,**(default sampling)**, indicates the service was configured at the [**edit-forwarding-options sampling-level**] hierarchy level.
FPC Slot	Slot number of the FPC for which the flow information is displayed.
Local interface index	Index counter of the local interface.
Interface state	Accounting state of the passive monitoring interface. • **Accounting**—PIC is actively accounting. • **Disabled**—PIC has been disabled from the CLI. • **Not accounting**—PIC is up but not accounting. This can happen while the PIC is coming online, or when the PIC is up but has no logical unit configured under the physical interface. • Unknown
Group index	Integer that represents the monitoring group of which the PIC is a member. **Group index** is a mapping from the group name to an index. It is not related to the number of monitoring groups.
Export interval (in seconds)	Configured export interval for cflowd records, in seconds.
Export format	Configured export format.
Protocol	Protocol the PIC is configured to monitor.
Engine type	Configured engine type that is inserted in output cflowd packets.
Engine ID	Configured engine ID that is inserted in output cflowd packets.
Route Record Count	Number of routes recorded.

Table 129: show services accounting status Output Fields *(continued)*

Field	Field Description
AS Record Count	Number of autonomous systems recorded.
Route Records Set	Status of route recording; whether routes are recorded or not.
Configuration Set	Status of monitoring configuration; whether monitoring configuration is set or not.

Sample Output

show services accounting status name (Monitoring PIC Interface)

`user@host>` **show services accounting status name count1**

```
Service Accounting interface: mo-2/0/0, Local interface index: 468
Service name: count1
Interface state: Accounting
  Group index: 0
  Export interval (in seconds): 60, Export format: cflowd v8
  Protocol: IPv4, Engine type: 55, Engine ID: 5
```

Sample Output

show services accounting status name (Service PIC Interface)

`user@host>` **show services accounting status name**

```
Service Accounting interface: sp-0/1/0
Interface state: Accounting
  Export format: 9, Route record count: 0
  IFL to SNMP index count: 7, AS count: 0
  Configuration set: Yes, Route record set: No, IFL SNMP map set: Yes
```

```
Service Accounting interface: sp-1/0/0
Interface state: Accounting
  Export format: 9, Route record count: 33
```

```
  IFL to SNMP index count: 7, AS count: 1
  Configuration set: Yes, Route record set: Yes, IFL SNMP map set: Yes
```

show services accounting status inline-jflow fpc-slot (When IPv4, IPv6 and Bridge Family Are Configured)

```
user@host> show services accounting status inline-jflow fpc-slot 0
```

```
FPC Slot: 0
     IPV4 export format: Version-IPFIX, IPV6 export format: Not set
     BRIDGE export format: Version-IPFIX, MPLS export format: Version-IPFIX
     IPv4 Route Record Count: 31, IPv6 Route Record Count: 0, MPLS Route Record
Count: 13
     Route Record Count: 44, AS Record Count: 1
     Route-Records Set: Yes, Config Set: Yes
     Service Status: PFE-0: Steady PFE-1: Steady
     Using Extended Flow Memory?: PFE-0: No PFE-1: No
     Flex Flow Sizing ENABLED?: PFE-0: No PFE-1: No
     IPv4 MAX FLOW Count: 1024, IPv6 MAX FLOW Count: 512
     BRIDGE MAX FLOW Count: 1024, MPLS MAX FLOW Count: 1024
     MAX Flow Table size: 15
```

show services accounting status inline-jflow (MX80 Router When Both IPv4 and IPv6)

```
user@host> show services accounting status inline-jflow
```

```
Status information
    TFEB Slot: 0
    Export format: IP-FIX
    IPv4 Route Record Count: 6, IPv6 Route Record Count: 8
    Route Record Count: 14, AS Record Count: 1
    Route-Records Set: Yes, Config Set: Yes
```

show services accounting status inline-jflow fpc-slot (PTX1000 Router When Both IPv4 and IPv6 Are Configured)

```
user@host> show services accounting status inline-jflow fpc-slot 0
```

```
Status information
FPC Slot: 0
IPV4 export format: Version-IPFIX, IPV6 export format: Version-IPFIX
MPLS export format: Not set
IPv4 Route Record Count: 23, IPv6 Route Record Count: 3, MPLS Route Record Count:
```

```
 0
Route Record Count: 26, AS Record Count: 1
Route-Records Set: Yes, Config Set: Yes
```

show services accounting status inline-jflow (SRX Series Devices When Both IPv4 and IPv6 Are Configured)

```
user@host> show services accounting status inline-jflow
```

```
Status information
    FPC Slot: 0
    IPV4 export format: Version9, IPV6 export format: Version9
    BRIDGE export format: Not set, MPLS export format: Not set
    IPv4 Route Record Count: 24, IPv6 Route Record Count: 0, MPLS Route Record
Count: 0
    Route Record Count: 24, AS Record Count: 1
    Route-Records Set: Yes, Config Set: Yes
    Service Status: PFE-0: Steady
    Using Extended Flow Memory?: PFE-0: No
    Flex Flow Sizing ENABLED?: PFE-0: No
    IPv4 MAX FLOW Count: 0, IPv6 MAX FLOW Count: 0
    BRIDGE MAX FLOW Count: 0, MPLS MAX FLOW Count: 0
```

show services accounting usage

Syntax

```
show services accounting usage
<name service-name>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

Display the CPU usage of PIC used for active flow monitoring.

Options

none—Display CPU usage for all service names.

name *service-name*—(Optional) Display CPU usage for the specified service name.

Additional Information

When no route record has been downloaded from the PIC, this command reports no flows, even though packets are being sampled.

Required Privilege Level

view

List of Sample Output

show services accounting usage (Monitoring PIC Interface) on page 1323
show services accounting usage (Service PIC Interface) on page 1323

Output Fields

Table 130 on page 1322 lists the output fields for the **show services accounting usage** command. Output fields are listed in the approximate order in which they appear.

Table 130: show services accounting usage Output Fields

Output Field	Output Field Description
Service Accounting interface	Name of the service accounting interface.
Service name	Name of a service that was configured at the [**edit-forwarding-options accounting**] hierarchy level. The default display, **(default sampling)**, indicates the service was configured at the [**edit-forwarding-options sampling-level**] hierarchy level.
Local interface index	Index counter of the local interface.

Table 130: show services accounting usage Output Fields *(continued)*

Output Field	Output Field Description
Uptime	Time that the PIC has been operational (in milliseconds).
Interrupt time	Total time that the PIC has spent processing packets since the last PIC reset (in microseconds).
Load (5 second)	CPU load on the PIC, averaged more than 5 seconds. The number is a percentage obtained by dividing the time spent on active tasks by the total elapsed time.
Load (1 minute)	CPU load on the PIC, averaged more than 1 minute. The number is a percentage obtained by dividing the time spent on active tasks by the total elapsed time.

Sample Output

show services accounting usage (Monitoring PIC Interface)

`user@host>` **show services accounting usage**

```
Service Accounting interface: mo-1/1/0, Local interface index: 15
Service name: (default sampling)
  CPU utilization
    Uptime: 600413856 milliseconds, Interrupt time: 2403 microseconds
    Load (5 second): 43%, Load (1 minute): 24%
```

show services accounting usage (Service PIC Interface)

`user@host>` **show services accounting usage**

```
Service Accounting interface: sp-0/1/0
Service name: (default sampling)
  CPU utilization
    Uptime: 7853940 milliseconds, Interrupt time: 0 microseconds
    Load (5 second): 2%, Load (1 minute): 0%
```

```
Service Accounting interface: sp-0/1/0
Service name: (default sampling)
  CPU utilization
```

```
    Uptime: 331160 milliseconds, Interrupt time: 0 microseconds
    Load (5 second): 2%, Load (1 minute): 0%
```

show services dynamic-flow-capture content-destination

Syntax

```
show services dynamic-flow-capture content-destination capture-group group-name destination-identifier identifier
<terse>
```

Release Information

Command introduced in Junos OS Release 7.4.

Description

(M320 Series routers and T Series routers only) Display information about the content destination that receives packets from the dynamic flow capture (DFC) interface.

Options

capture-group *group-name*—Display information for the specified capture-group identifier.

destination-identifier *identifier*—Display information for the specified content destination identifier.

terse—(Optional) Display summary information.

Required Privilege Level

view

List of Sample Output

show services dynamic-flow-capture content-destination capture-group on page 1326

Output Fields

Table 131 on page 1325 lists the output fields for the **show services dynamic-flow-capture content-destination** command. Output fields are listed in the approximate order in which they appear.

Table 131: show services dynamic-flow-capture content-destination Output Fields

Output Field	Output Field Description
Capture group	Name of the capture group.
Content destination	Name of the content destination.
Criteria	Number of criteria specified.
Bandwidth	Bandwidth used by the matched traffic.
Matched packets	Number of matched packets sent to the content destination.

Table 131: show services dynamic-flow-capture content-destination Output Fields *(continued)*

Output Field	Output Field Description
Matched bytes	Number of matched bytes sent to the content destination.
Congestion notifications	Number of notification messages sent.

Sample Output

show services dynamic-flow-capture content-destination capture-group

`user@host>` **show services dynamic-flow-capture content-destination capture-group g1 destination-identifier cd1 terse**

```
  Capture group: g1, Content destination: cd1, Criteria: 0, Bandwidth: 0, Matched
packets: 0, Matched bytes: 0, Congestion notifications: 0
```

show services dynamic-flow-capture control-source

Syntax

```
show services dynamic-flow-capture control-source capture-group group-name control-source source-identifier
   identifier
<detail | terse>
```

Release Information

Command introduced in Junos OS Release 7.4.

Description

(M320 Series routers and T Series routers only) Display information about the control source that makes dynamic flow capture requests to the dynamic flow capture interface.

Options

capture-group *group-name*—Capture group identifier.

source-identifier *identifier*—Control source identifier.

detail | terse—(Optional) Display the specified level of output.

Required Privilege Level

view

List of Sample Output

show services dynamic-flow-capture control-source source-identifier capture-group on page 1328
show services dynamic-flow-capture control-source ource-identifier capture-group detail on page 1329

Output Fields

Table 132 on page 1327 lists the output fields for the **show services dynamic-flow-capture control-source** scommand. Output fields are listed in the approximate order in which they appear.

Table 132: show services dynamic-flow-capture control-source Output Fields

Output Field	Output Field Description
Capture group	Name of the capture group.
Control source	Name of the control source.
Criteria added, Criteria add failed	Number of criteria added or added and failed.
Active criteria	Number of active criteria.

Table 132: show services dynamic-flow-capture control-source Output Fields *(continued)*

Output Field	Output Field Description
Static criteria, Dynamic criteria	Number of static or dynamic criteria.
Control protocol requests	Total number of control protocol requests.
Requests	Number of **Add**, **Delete**, **List**, **Refresh**, and **No-op** control protocol requests.
Failed	Number of **Add**, **Delete**, **List**, **Refresh**, and **No-op** failed control protocol requests.
Add request rate	Rate of add requests.
Add request peak rate	Peak rate of add requests.
Bandwidth across all criteria	Bandwidth used by all the requests.
Total notifications	Total number of notifications sent and the number of notifications by category: **Restart**, **Rollover**, **Timeout**, **Congestion**, **Congestion delete**, and **Dups** (duplicates) dropped.
Criteria deleted	Total number of criteria deleted and the number of deleted criteria by category: **Timeout idle**, **Timeout total**, **Packets**, and **Bytes**.
Sequence number	Sequence number.

Sample Output

show services dynamic-flow-capture control-source source-identifier capture-group

`user@host>` **show services dynamic-flow-capture control-source source-identifier cs0_cg0 capture-group cg_0**

```
Capture group: cg_0, Control source: cs0_cg0
 Criteria added: 28, Criteria add failed: 0, Active criteria: 0, Control protocol
 requests: 28,  Add request rate: 0,
```

```
 Add request peak rate: 1, Bandwidth across all criteria: 0, Total notifications:
 1, Criteria deleted: 28, Sequence number: 0
```

show services dynamic-flow-capture control-source ource-identifier capture-group detail

user@host> **show services dynamic-flow-capture control-source source-identifier cs0_cg0 capture-group cg_0 detail**

```
Capture group: cg_0, Control source: cs0_cg0
  Criteria added: 28, Criteria add failed: 0
  Active criteria: 0
    Static criteria: 0, Dynamic criteria: 0
  Control protocol requests: 28
                    Add        Delete          List       Refresh         No-op
    Requests         28             0             0             0             0
    Failed            0             0             0             0             0
  Add request rate: 0
  Add request peak rate: 1
  Bandwidth across all criteria: 0
  Total notifications: 1
    Restart: 1, Rollover: 0, No-op: 0, Timeout: 0, Congestion: 0, Congestion delete:
 0, Dups dropped: 0
  Criteria deleted: 28
    Timeout idle: 0, Timeout total: 0, Packets: 0, Bytes: 0
  Sequence number: 0
```

show services dynamic-flow-capture statistics

Syntax

```
show services dynamic-flow-capture statistics capture-group group-name
```

Release Information

Command introduced in Junos OS Release 7.4.

Description

(M320 Series routers and T Series routers only) Display statistics information about the capture group specified for dynamic flow capture.

Options

capture-group *group-name*—Display information for the specified capture group identifier.

Required Privilege Level

view

List of Sample Output

show services dynamic-flow-capture statistics capture-group on page 1332

Output Fields

Table 133 on page 1330 lists the output fields for the **show services dynamic-flow-capture statistics** command. Output fields are listed in the approximate order in which they appear.

Table 133: show services dynamic-flow-capture statistics Output Fields

Output Field	Output Field Description
Input	Incoming dynamic flow capture packet statistics: • **Control protocol packets**—Number of control protocol packets received. • **Captured data packets**—Number of data packets captured. • **Control IRI packets**—Number of control IRI packets received.

Table 133: show services dynamic-flow-capture statistics Output Fields *(continued)*

Output Field	Output Field Description
Control protocol drops	Control protocol packets dropped for the following reasons: • **Not IP packets**—Dropped packets were not IP packets. • **Not UDP packets**—Dropped packets were not User Datagram Protocol (UDP) packets. • **Invalid destination address**—Dropped packets had invalid destination addresses. • **No memory**—Packets dropped because of insufficient memory. • **Unauthorized control source**—Packets dropped because the control source was not authenticated. • **Bad request**—Packets dropped because the request was invalid. • **Unknown control source**—Packets dropped because the control source was not known. • **Not DTCP**—Dropped packets did not adhere to the control protocol format. • **Bad command line**—Packets dropped because of a version mismatch. • **Bandwidth exceeded**—Packets dropped because the bandwidth was exceeded. • **Drop rate due to exceeded bandwidth**—Rate of traffic dropped because the bandwidth was exceeded. • **Other**—Packets dropped for other reasons or undetermined causes.
Input drops	Incoming dynamic flow capture packets dropped for the following reasons: • **Unknown packets**—Packets dropped because the packet type was not recognized. • **Captured data not IPv4**—Packets dropped because they were not IPv4 packets. • **Captured data too small**—Packets dropped because they were smaller than the size reported in their headers. • **Captured data drops**—Data packets dropped because of undetermined causes. • **Captured data not matched**—Packets dropped because they did not match filter criteria. • **Bandwidth exceeded**—Packets dropped because the bandwidth was exceeded. • **Drop rate due to exceeded bandwidth**—Rate of traffic dropped because the bandwidth was exceeded.
Output	Outgoing dynamic flow capture packet statistics: • **Control protocol packets**—Number of control protocol packets sent. • **Captured data packets**—Number of captured data packets sent.
Output drops	Outgoing packets dropped: • **Control protocol drops**—Number of control protocol packets dropped. • **Captured data drops**—Number of captured data packets dropped.

Table 133: show services dynamic-flow-capture statistics Output Fields *(continued)*

Output Field	Output Field Description
Flow Statistics	DFC flow statistics: • **Active flow cache entries** • **Active flow cache usage percentage** • **Flow cache entries allocated** • **Number of control sources** • **Number of content destinations** • **Number of criteria** • **Maximum criteria matching one flow** • **Cached flows purged for memory** • **Maximum filters matching one packet**

Sample Output

show services dynamic-flow-capture statistics capture-group

`user@host>` **show services dynamic-flow-capture statistics capture-group g1**

```
Input:

  Control protocol packets: 643, Captured data packets: 69977, Control IRI packets:
 337

Control protocol drops:

  Not IP packets: 0, Not UDP packets: 3, Invalid destination address: 0, No memory:
 0, Unauthorized control source: 0,

  Bad request: 0, Unknown control source: 0, Not DTCP: 0, Bad command line: 0,
Bandwidth exceeded: 0,

  Drop rate due to exceeded bandwidth: 0, Other: 0

Input drops:

  Unknown packets: 0, Captured data not IPv4: 0, Captured data too small: 0,
Captured data drops: 0, Captured data not matched: 0,
```

```
  Bandwidth exceeded: 0, Drop rate due to exceeded bandwidth: 0

Output:

  Control protocol packets: 644, Captured data packets: 1119624

Output drops:

  Control protocol drops: 0, Captured data drops: 0

Flow Statistics:

  Active flow cache entries: 40, Active flow cache usage percentage: 0, Flow cache
 entries allocated: 40,

  Number of control sources: 4, Number of content destinations: 64, Number of
criteria: 640,

  Maximum criteria matching one flow: 16, Cached flows purged for memory: 0, Maximum
 filters matching one packet: 16
```

show services flow-collector file interface

Syntax

```
show services flow-collector file interface (all | cp-fpc/pic/port)
<detail | extensive | terse>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display information about flow collector files.

Options

none—Display file information for all configured flow collector interfaces.

all | cp-*fpc/pic/port*—Display file information for all configured flow collector interfaces or for the specified interface.

detail | extensive | terse—(Optional) Display the specified level of output.

Additional Information

No entries are displayed for files that have been successfully transferred.

Required Privilege Level

view

List of Sample Output

show services flow-collector file interface extensive on page 1335

Output Fields

Table 134 on page 1334 lists the output fields for the **show services flow-collector file interface** command. Output fields are listed in the approximate order in which they appear.

Table 134: show services flow-collector file interface Output Fields

Output Field	Output Field Description	Level of Output
Filename	Name of the file created on the flow collector interface.	All levels
Flows	Total number of collector flows for which records are present in the file.	none specified

Table 134: show services flow-collector file interface Output Fields *(continued)*

Output Field	Output Field Description	Level of Output

Throughput	Throughput statistics:
	• **Flow records**—Number of flow records in the file.
	  ◦ **per second**—Average number of flow records per second.
	  ◦ **peak per second**—Peak number of flow records per second.
	• **Uncompressed bytes**—Total file size before compression.
	  ◦ **per second**—Average number of uncompressed bytes per second.
	  ◦ **peak per second**—Peak number of uncompressed bytes per second.
	• **Compressed bytes**—Total file size after compression.
	  ◦ **per second**—Average number of compressed bytes per second.
	  ◦ **peak per second**—Peak number of compressed bytes per second.
		extensive

Status	File statistics:
	• **Compressed blocks**—(**extensive** output only) Data blocks in the file that have been compressed. The file is exported only when the compressed block count and block count become the same.
	• **Block count**—(**extensive** output only) Total number of data blocks in the file.
	• **State**—Processing state of the file.
	  ◦ **Active**—The flow collector interface is writing to the file.
	  ◦ **Export 1**—File export is in progress to the primary server.
	  ◦ **Export 2**—File export is in progress to the secondary server.
	  ◦ **Wait**—File is pending export.
	  ◦ **Transfer attempts 0**.—Number of attempts made to transfer the file. If the file is successfully transferred in the first attempt, this field is 0.
		All levels

Sample Output

show services flow-collector file interface extensive

`user@host>` **show services flow-collector file interface cp-3/2/0 extensive**

```
Filename: cFlowd-py69Ni69-0-20031112_014301-so_3_0_0_0.bcp.bi.gz
  Throughput:
    Flow records: 188365, per second: 238, peak per second: 287
```

```
  Uncompressed bytes: 21267756, per second: 27007, peak per second: 32526
  Compressed bytes: 2965643, per second: 0, peak per second: 22999
Status:
  Compressed blocks: 156, Block count: 156
  State: Active, Transfer attempts: 0
```

show services flow-collector input interface

Syntax

```
show services flow-collector input interface (all | cp-fpc/pic/port)
<detail | extensive | terse>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display the number of packets received by collector interfaces from monitoring interfaces.

Options

none—Display packets received by all configured flow collector interfaces.

all | cp-*fpc/pic/port*—Display packets received by all configured flow collector interfaces or by the specified interface.

detail | extensive | terse—(Optional) Display the specified level of output.

Required Privilege Level

view

List of Sample Output

show services flow-collector input interface on page 1338
show services flow-collector input interface all on page 1338

Output Fields

Table 135 on page 1337 lists the output fields for the **show services flow-collector input interface** command. Output fields are listed in the approximate order in which they appear.

Table 135: show services flow-collector input interface Output Fields

Output Field	Output Field Description
Interface	Name of the monitoring interface.
Packets	Number of packets traveling from the monitoring interface to the flow collector interface.
Bytes	Number of bytes traveling from the monitoring interface to the flow collector interface.

Sample Output

show services flow-collector input interface

```
user@host> show services flow-collector input interface cp-3/2/0
```

```
Interface                      Packets        Bytes
mo-3/0/0.0                       21706     32328568
mo-3/1/0.0                       21706     32329096
```

show services flow-collector input interface all

```
user@host> show services flow-collector input interface all
```

```
Flow collector interface: cp-6/1/0
Interface state: Collecting flows
Interface                      Packets        Bytes
mo-3/0/0.0                         274       416232
mo-3/3/0.0                         274       416184
mo-1/0/0.0                         274       416232
mo-1/1/0.0                         274       416232
mo-1/2/0.0                         274       416232
mo-1/3/0.0                         274       416232
mo-3/1/0.0                         274       416232
mo-4/0/0.0                         274       416232
mo-4/1/0.0                         274       416232
mo-4/2/0.0                         274       416184
mo-4/3/0.0                         274       416232
mo-5/0/0.0                         274       416232
mo-5/1/0.0                         274       416232
mo-5/2/0.0                         274       416232
mo-5/3/0.0                         274       416232
mo-6/0/0.0                         274       416232

Flow collector interface: cp-6/3/0
Interface state: Collecting flows
```

show services flow-collector interface

Syntax

```
show services flow-collector interface (all | cp-fpc/pic/port)
<detail | extensive | terse>
```

Release Information

Command introduced before Junos OS Release 7.4.

Description

(M40e, M160, and M320 Series routers and T Series routers only) Display overall statistics for the flow collector application.

Options

none—Display statistics for flow collector applications on all interfaces.

all | cp-*fpc/pic/port*—Display statistics for flow collector applications on all interfaces or for the specified interface.

detail | extensive | terse—(Optional) Display the specified level of output.

Required Privilege Level

view

List of Sample Output

show services flow-collector interface all detail on page 1343
show services flow-collector interface all extensive on page 1344
show services flow-collector interface all terse on page 1346
show services flow-collector interface extensive on page 1347

Output Fields

Table 136 on page 1339 lists the output fields for the **show services flow-collector interface** command. Output fields are listed in the approximate order in which they appear.

Table 136: show services flow-collector interface Output Fields

Output Field	Output Field Description	Level of Output
Flow collector interface	Name of the flow collector interface.	All levels
Interface state	Collecting flow state for the interface.	All levels
Packets	Total number of packets received.	none specified

Table 136: show services flow-collector interface Output Fields *(continued)*

Output Field	Output Field Description	Level of Output
Flows Uncompressed Bytes	Total uncompressed data size for all files created on this PIC.	none specified
Compressed Bytes	Total compressed data size for all files created on this PIC.	none specified
FTP bytes	Total number of bytes transferred to the FTP server, including those dropped during transfer.	none specified
FTP files	Total number of FTP transfers attempted by the server.	none specified
Memory	Bytes used on the PIC and bytes free.	**detail extensive**

Input	Incoming flow collector packet statistics:
	• **Packets**—Number of packets received on the unit.
	  ◦ **per second**—Average number of packets per second.
	  ◦ **peak per second**—Peak number of packets per second.
	• **Bytes**—Number of bytes received on the unit.
	  ◦ **per second**—Average number of bytes per second.
	  ◦ **peak per second**—Peak number of bytes per second.
	• **Flow records processed**—Number of records in the flow collector packets that were processed by the flow-collector interface.
	  ◦ **per second**—Average number of flow records processed per second.
	  ◦ **peak per second**—Peak number of flow records per second.
		detail extensive

Table 136: show services flow-collector interface Output Fields *(continued)*

Output Field	Output Field Description	Level of Output

Allocation	Data block statistics:
	• **Blocks allocated**—Total number of data blocks (containing flow records) allocated to the files created on this PIC.
	  ◦ **per second**—Average number of blocks allocated per second.
	  ◦ **peak per second**—Peak number of blocks allocated per second.
	• **Blocks freed**—Total number of data blocks freed.
	  ◦ per second—Average number of blocks freed per second.
	  ◦ peak per second—Peak number of blocks freed per second.
	• **Blocks unavailable**—Total number of data block requests denied, typically because of a memory shortage.
	  ◦ per second—Average number of blocks unavailable per second.
	  ◦ peak per second—Peak number of blocks unavailable per second.
		extensive

Files	File statistics, incremented since the PIC last booted: • **Files created**—Total number of files created on this PIC. • **Files exported**— Number of files successfully created and exported. • **Files destroyed**—(**extensive** output only) Number of files successfully exported and files dropped by the flow collection interface.	**detail extensive**

Throughput	Throughput statistics:
	• **Uncompressed bytes**—Total uncompressed data size for all files created on this PIC.
	  ◦ **per second**—Average number of uncompressed bytes per second.
	  ◦ **peak per second**—Peak number of uncompressed bytes per second.
	• **Compressed bytes**—Total compressed data size for all files created on this PIC.
	  ◦ **per second**—Average number of compressed bytes per second.
	  ◦ **peak per second**—Peak number of compressed bytes per second.
		detail extensive

Table 136: show services flow-collector interface Output Fields *(continued)*

Output Field	Output Field Description	Level of Output
Packet drops	Number of packets dropped for the following causes: • **No memory**—Packets dropped because of insufficient memory. • **Not IP**—Packets dropped because they are not IP packets. • **Not IPv4**—Packets dropped because they are not IP version 4 packets. • **Too small**—Packets dropped because each packet was smaller than the size reported in its header. • **Fragments**—Packets dropped because of fragmentation. Fragments are not reassembled. • **ICMP**—Packets dropped because they are not ICMP packets. • **TCP**—Packets dropped because they are not TCP packets. • **Unknown**—Packets dropped because of undetermined causes. • **Not Junos flow**—Packets dropped because they are not interpreted by Junos OS. Junos OS interprets only IPv4, UDP cflowd version 5 packets.	**extensive**
File transfer	File transfer statistics: • **FTP bytes**—Total number of bytes transferred to the FTP server, including those dropped during transfer. • **FTP files**—Total number of FTP transfers attempted by the server. • **FTP failure**—Total number of FTP failures encountered by the server.	**detail extensive**
Flow collector interface	Physical interface acting as a flow collector.	**detail**

Table 136: show services flow-collector interface Output Fields *(continued)*

Output Field	Output Field Description	Level of Output

Export channel	Export channel **0** is unit 0. Export channel **1** is unit 1. Flow receive channel is unit 2. Server status statistics are the following:
	• **Current server Primary** or **Secondary**—Current FTP server being used. Value is
	• **Primary server state**—State of the server:
	  ◦ **OK**—Server is operating without problems.
	  ◦ **FTP error**—Server encountered an FTP protocol error while sending files.
	  ◦ **Network error**—Flow-collector interface has errors when contacting the primary FTP server.
	  ◦ **Unknown**—First file transfer has not been sent to the primary server.
	• **Secondary server state**—State of the server:
	  ◦ **OK**—Server is operating without errors.
	  ◦ **FTP error**—Server encountered an FTP protocol error while sending files.
	  ◦ **Network error**—Flow-collector interface has errors when contacting the secondary FTP server.
	  ◦ **Unknown**—First file transfer has not been sent to the secondary server.
	• **Not configured**—Secondary server is not configured.
		detail extensive

Sample Output

show services flow-collector interface all detail

`user@host>` **show services flow-collector interface all detail**

```
Flow collector interface: cp-6/1/0
Interface state: Collecting flows
Memory:
    Used: 51452732, Free: 440329088
Input:
    Packets: 4384, per second: 0, peak per second: 156
    Bytes: 6659616, per second: 0, peak per second: 249695
    Flow records processed: 131070, per second: 0, peak per second: 4914
Files:
```

```
    Files created: 1, per second: 0, peak per second: 0
    Files exported: 1, per second: 0, peak per second: 0
Throughput:
    Uncompressed bytes: 13742307, per second: 0, peak per second: 593564
    Compressed bytes: 3786177, per second: 0, peak per second: 162826
File Transfer:
    FTP bytes: 3786247, per second: 0, peak per second: 378620
    FTP files: 1, per second: 0, peak per second: 0
    FTP failure: 0
Export channel: 0
    Current server: Primary
    Primary server state: OK, Secondary server state: OK
Export channel: 1
    Current server: Primary
    Primary server state: Unknown, Secondary server state: OK

Flow collector interface: cp-6/3/0
Interface state: Collecting flows
Memory:
    Used: 51452732, Free: 440329088
Input:
    Packets: 0, per second: 0, peak per second: 0
    Bytes: 0, per second: 0, peak per second: 0
    Flow records processed: 0, per second: 0, peak per second: 0
Files:
    Files created: 0, per second: 0, peak per second: 0
    Files exported: 0, per second: 0, peak per second: 0
Throughput:
    Uncompressed bytes: 0, per second: 0, peak per second: 0
    Compressed bytes: 0, per second: 0, peak per second: 0
File Transfer:
    FTP bytes: 70, per second: 0, peak per second: 6
    FTP files: 0, per second: 0, peak per second: 0
    FTP failure: 0
Export channel: 0
    Current server: Primary
    Primary server state: Unknown, Secondary server state: OK
Export channel: 1
    Current server: Primary
    Primary server state: Unknown, Secondary server state: OK
```

show services flow-collector interface all extensive

`user@host>` **show services flow-collector interface all extensive**

```
Flow collector interface: cp-6/1/0
Interface state: Collecting flows
Memory:
    Used: 51452732, Free: 440329088
Input:
    Packets: 4384, per second: 0, peak per second: 156
    Bytes: 6659616, per second: 0, peak per second: 249695
    Flow records processed: 131070, per second: 0, peak per second: 4914
Allocation:
    Blocks allocated: 108, per second: 0, peak per second: 0
    Blocks freed: 108, per second: 0, peak per second: 10
    Blocks unavailable: 0, per second: 0, peak per second: 0
Files:
    Files created: 1, per second: 0, peak per second: 0
    Files exported: 1, per second: 0, peak per second: 0
    Files destroyed: 1, per second: 0, peak per second: 0
Throughput:
    Uncompressed bytes: 13742307, per second: 0, peak per second: 593564
    Compressed bytes: 3786177, per second: 0, peak per second: 162826
Packet drops:
    No memory: 0, Not IP: 0
    Not IPv4: 0, Too small: 0
    Fragments: 0, ICMP: 0
    TCP: 0, Unknown: 0
    Not JUNOS flow: 0
File Transfer:
    FTP bytes: 3786247, per second: 0, peak per second: 378620
    FTP files: 1, per second: 0, peak per second: 0
    FTP failure: 0
Export channel: 0
    Current server: Primary
    Primary server state: OK, Secondary server state: OK
Export channel: 1
    Current server: Primary
    Primary server state: Unknown, Secondary server state: OK

Flow collector interface: cp-6/3/0
Interface state: Collecting flows
Memory:
    Used: 51452732, Free: 440329088
Input:
    Packets: 0, per second: 0, peak per second: 0
    Bytes: 0, per second: 0, peak per second: 0
    Flow records processed: 0, per second: 0, peak per second: 0
```

```
Allocation:
    Blocks allocated: 0, per second: 0, peak per second: 0
    Blocks freed: 0, per second: 0, peak per second: 0
    Blocks unavailable: 0, per second: 0, peak per second: 0
Files:
    Files created: 0, per second: 0, peak per second: 0
    Files exported: 0, per second: 0, peak per second: 0
    Files destroyed: 0, per second: 0, peak per second: 0
Throughput:
    Uncompressed bytes: 0, per second: 0, peak per second: 0
    Compressed bytes: 0, per second: 0, peak per second: 0
Packet drops:
    No memory: 0, Not IP: 0
    Not IPv4: 0, Too small: 0
    Fragments: 0, ICMP: 0
    TCP: 0, Unknown: 0
    Not JUNOS flow: 0
File Transfer:
    FTP bytes: 70, per second: 0, peak per second: 6
    FTP files: 0, per second: 0, peak per second: 0
    FTP failure: 0
Export channel: 0
    Current server: Primary
    Primary server state: Unknown, Secondary server state: OK
Export channel: 1
    Current server: Primary
    Primary server state: Unknown, Secondary server state: OK
```

show services flow-collector interface all terse

user@host> **show services flow-collector interface all terse**

```
Flow collector interface: cp-6/1/0
Interface state: Collecting flows
  Packets     Bytes     Flows Uncompressed  Compressed     FTP bytes FTP files
                                    Bytes       Bytes
     4384   6659616    131070     13742307     3786177       3786247         1

Flow collector interface: cp-6/3/0
Interface state: Collecting flows
  Packets     Bytes     Flows Uncompressed  Compressed     FTP bytes FTP files
                                    Bytes       Bytes
        0         0         0            0           0            70         0
```

show services flow-collector interface extensive

user@host> **show services flow-collector interface cp-5/2/0 extensive**

```
Flow collector interface: cp-5/2/0
Interface state: Collecting flows
Memory:
     Used: 458311860, Free: 40810008
Input:
     Packets: 922629, per second: 2069, peak per second: 3266
     Bytes: 1376559252, per second: 3096940, peak per second: 4880051
     Flow records processed: 25764957, per second: 42564, peak per second: 98124
Allocation:
     Blocks allocated: 20862, per second: 31, peak per second: 72
     Blocks freed: 17161, per second: 40, peak per second: 202
     Blocks unavailable: 58786, per second: 652, peak per second: 1120
Files:
     Files created: 52, per second: 0, peak per second: 0
     Files exported: 42, per second: 0, peak per second: 0
     Files destroyed: 42, per second: 0, peak per second: 0
Throughput:
     Uncompressed bytes: 2592070401, per second: 7297307,
     peak per second: 8630023
     Compressed bytes: 659600068, per second: 1858458, peak per second: 2198471
Packet drops:
     No memory: 58786, Not IP: 0
     Not IPv4: 0, Too small: 0
     Fragments: 0, ICMP: 0
     TCP: 0, Unknown: 0
     Not JUNOS flow: 0
File Transfer:
     FTP bytes: 585981447, per second: 1313320, peak per second: 4857798
     FTP files: 48, per second: 0, peak per second: 0
     FTP failure: 8
Export channel: 0
     Current server: Primary
    Primary server state: FTP error, Secondary server state: Not configured Export
 channel: 1
     Current server: Primary
     Primary server state: OK, Secondary server state: Not configured
```

show services inline-monitoring statistics fpc-slot

Syntax

```
show services inline-monitoring statistics fpc-slot fpc-slot
```

Release Information

Command introduced in Junos OS Release 19.4R1 on MX Series routers with MPCs excluding MPC10E and MPC11E linecards.

Description

Display inline-monitoring statistical information.

Options

fpc-slot—Display inline-monitoring statistical information for the specified Flexible PIC Concentrator (FPC) number.

Range: 0 through 11

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Inline Monitoring Services | 362

List of Sample Output

show services inline-monitoring services fpc-slot on page 1349

Output Fields

Table 137 on page 1348 lists the output fields for the **show services inline-monitoring statistics fpc-slot** command. Output fields are listed in the approximate order in which they appear.

Table 137: show services inline-monitoring statistics fpc-slot Output Fields

Field Name	Field Description
FPC Slot	Flexible PIC Concentrator (FPC) slot level counters: • Packets—Number of packets services under FPC slot. • Bytes—Number of bytes services under FPC slot.

Table 137: show services inline-monitoring statistics fpc-slot Output Fields *(continued)*

Field Name	Field Description
Instance Name	Instance level counters: • Packets—Number of packets services under instance number under the FPC slot. • Bytes—Number of bytes services under instance number under the FPC slot.
Collector Name	Collector level counters: • Packets—Number of packets services under collector number of an instance under the FPC slot. • Bytes—Number of bytes services under collector number of an instance under the FPC slot.

Sample Output

show services inline-monitoring services fpc-slot

`user@host>` **show services inline-monitoring services fpc-slot *fpc-slot***

```
IMON Statistics
   FPC Slot     :  <FPC_SLOT>
   Packets      :  <F_P>       Bytes  :  <F_B>

     Instance Name  :  <I1>
     Packets  :  <I1_P>      Bytes  :  <I1_B>

   Collector Name  :  <I1_C1>
   Packets  :  <I1_C1_P>    Bytes  :  <I1_C1_B>

   Collector Name  :  <I1_C2>
   Packets  :  <I1_C2_P>    Bytes  :  <I1_C2_B>

     Instance Name  :  <I2>
     Packets  :  <I2_P>      Bytes  :  <I2_B>

   Colector Name  :  <I2_C1>
   Packets  :  <I2_C1_P>    Bytes  :  <I2_C1_B>
```

show services rpm active-servers

Syntax

```
show services rpm active-servers
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Command introduced in Junos OS Release 13.2 for PTX Series Packet Transport routers.
Command introduced in Junos OS Release 18.1 for QFX Series switches.

Description

Display the protocols and corresponding ports for which a router or switch is configured as a real-time performance monitoring (RPM) server.

Options

This command has no options.

Required Privilege Level

view

List of Sample Output

show services rpm active-servers on page 1351

Output Fields

Table 138 on page 1350 lists the output fields for the **show services rpm active-servers** command. Output fields are listed in the approximate order in which they appear.

Table 138: show services rpm active-servers Output Fields

Field Name	Field Description
Protocol	Protocol configured on the receiving probe server. The protocol can be the User Datagram Protocol (UDP) or the Transmission Control Protocol (TCP).
Port	Port configured on the receiving probe server.
Destination interface name	Output interface name for the probes.

Sample Output

show services rpm active-servers

user@host> **show services rpm active-servers**

```
    Protocol: TCP, Port: 50000, Destination interface name: lt-0/0/0.0
    Protocol: UDP, Port: 50001, Destination interface name: lt-0/0/0.0
```

show services rpm history-results

Syntax

```
show services rpm history-results
<brief | detail>
<dst-interface interface-name>
<owner owner>
<limit number>
<since time>
<source-address address>
<target-address address>
<test name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 on EX Series switches.
Command introduced in Junos OS Release 13.2 on PTX Series Packet Transport routers.
dst-interface, **limit**, **source-address**, and **target-address** options introduced in Junos OS Release 18.1R1 on MX Series.
owner and **test** options became optional in Junos OS Release 18.1R1 on MX Series.
Command introduced in Junos OS Release 18.1 on QFX Series switches.

Description

Display the results stored for the specified real-time performance monitoring (RPM) probes.

Options

none—(Optional) Display the results of the last 50 probes for all RPM instances.

brief | detail—(Optional) Display the specified level of output.

dst-interface *interface-name*—(Optional) Display information only for RPM probes that are generated on this MS-MPC or MS-MIC services interface. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface. You must also configure the **owner** option.

limit *number*—(Optional) Limit the number of results that are displayed. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface. You must also configure the **owner** option.

Range: 1 through 4,294,967,295

Default: 100

owner *owner*—(Optional) Display information only for probes with the specified probe owner. You must configure **owner** if you configure any of the following options: **dst-interface**, **limit**, **source-address**, or **target-address**.

since *time*—(Optional) Display information from the specified time. Specify time as ***yyyy-mm-dd.hh:mm:ss***.

source-address *address*—(Optional) Display information only for probes with the specified source address. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface. You must also configure the **owner** option.

target-address *address*—(Optional) Display information only for probes with the specified target address. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface. You must also configure the **owner** option.

test *name*—(Optional starting in Junos OS Release 18.1R1) Display information only for the specified test.

Do not configure **test** if you configure any of the following options: **dst-interface**, **limit**, **source-address**, or **target-address**. These options do not work when you configure **test**.

Required Privilege Level
view

List of Sample Output
show services rpm history-results owner test on page 1354
show services rpm history-results owner test detail on page 1355

Output Fields
Table 139 on page 1353 lists the output fields for the **show services rpm history-results** command. Output fields are listed in the approximate order in which they appear.

Table 139: show services rpm history-results Output Fields

Field Name	Field Description	Level of Output
Owner	Probe owner.	All levels
Test	Name of a test for a probe instance.	All levels
Probe received	Timestamp when the probe result was determined.	All levels
Round trip time	Average ping round-trip time (RTT), in microseconds.	All levels
Probe results	Result of a particular probe performed by a remote host. The following information is contained in the results: • **Response received**—Timestamp when the probe result was determined. • **Rtt**—Average ping round-trip time (RTT), in microseconds.	**detail**

Table 139: show services rpm history-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Results over current test	Displays the results for the current test by probe at the time each probe was completed, as well as the status of the current test at the time the probe was completed.	**detail**
Probes sent	Number of probes sent with the current test.	**detail**
Probes received	Number of probe responses received within the current test.	**detail**
Loss percentage	Percentage of lost probes for the current test.	**detail**
Measurement	Increment of measurement. Possible values are round-trip time delay and, for the probe type icmp-pin-timestamp, the egress and ingress delay: • **Minimum**—Minimum RTT, ingress delay, or egress delay measured over the course of the current test. • **Maximum**—Maximum RTT, ingress delay, or egress delay measured over the course of the current test. • **Average**—Average RTT, ingress delay, or egress delay measured over the course of the current test. • **Jitter**—Difference, in microseconds, between the maximum and minimum RTT measured over the course of the current test. • **Stddev**—Standard deviation of the round-trip time, in microseconds, measured over the course of the current test.	**detail**

Sample Output

show services rpm history-results owner test

```
user@host> show services rpm history-results owner p1 test t1

     Owner, Test                 Probe received              Round trip time
    p1, t1                     Wed Aug 12 01:02:35 2009             315 usec
    p1, t1                     Wed Aug 12 01:02:36 2009             266 usec
    p1, t1                     Wed Aug 12 01:02:37 2009             314 usec
    p1, t1                     Wed Aug 12 01:02:38 2009             388 usec
```

```
p1, t1                      Wed Aug 12 01:02:39 2009             316 usec
p1, t1                      Wed Aug 12 01:02:40 2009             271 usec
p1, t1                      Wed Aug 12 01:02:41 2009             314 usec
p1, t1                      Wed Aug 12 01:02:42 2009            1180 usec
```

show services rpm history-results owner test detail

user@host> **show services rpm history-results owner p1 test t1 detail**

```
Owner: p1, Test: t1, Probe type: icmp-ping-timestamp
Probe results:
  Response received, Wed Aug 12 01:02:35 2009,
  Client and server hardware timestamps
  Rtt: 315 usec
Results over current test:
  Probes sent: 1, Probes received: 1, Loss percentage: 0
  Measurement: Round trip time
    Samples: 1, Minimum: 315 usec, Maximum: 315 usec, Average: 315 usec,
    Peak to peak: 0 usec, Stddev: 0 usec, Sum: 315 usec

Owner: p1, Test: t1, Probe type: icmp-ping-timestamp
Probe results:
  Response received, Wed Aug 12 01:02:36 2009,
  Client and server hardware timestamps
  Rtt: 266 usec, Round trip jitter: -50 usec,
  Round trip interarrival jitter: 3 usec
Results over current test:
  Probes sent: 2, Probes received: 2, Loss percentage: 0
  Measurement: Round trip time
    Samples: 2, Minimum: 266 usec, Maximum: 315 usec, Average: 291 usec,
    Peak to peak: 49 usec, Stddev: 24 usec, Sum: 581 usec
  Measurement: Negative round trip jitter
    Samples: 1, Minimum: 50 usec, Maximum: 50 usec, Average: 50 usec,
    Peak to peak: 0 usec, Stddev: 0 usec, Sum: 50 usec

Owner: p1, Test: t1, Probe type: icmp-ping-timestamp
Probe results:
  Response received, Wed Aug 12 01:02:37 2009,
  Client and server hardware timestamps
  Rtt: 314 usec, Round trip jitter: 49 usec,
  Round trip interarrival jitter: 6 usec
Results over current test:
  Probes sent: 3, Probes received: 3, Loss percentage: 0
```

```
  Measurement: Round trip time
    Samples: 3, Minimum: 266 usec, Maximum: 315 usec, Average: 298 usec,
    Peak to peak: 49 usec, Stddev: 23 usec, Sum: 895 usec
  Measurement: Positive round trip jitter
    Samples: 1, Minimum: 49 usec, Maximum: 49 usec, Average: 49 usec,
    Peak to peak: 0 usec, Stddev: 0 usec, Sum: 49 usec
  Measurement: Negative round trip jitter
    Samples: 1, Minimum: 50 usec, Maximum: 50 usec, Average: 50 usec,
    Peak to peak: 0 usec, Stddev: 0 usec, Sum: 50 usec

Owner: p1, Test: t1, Probe type: icmp-ping-timestamp
Probe results:
  Response received, Wed Aug 12 01:02:38 2009,
  Client and server hardware timestamps
  Rtt: 388 usec, Round trip jitter: 74 usec,
  Round trip interarrival jitter: 10 usec
Results over current test:
  Probes sent: 4, Probes received: 4, Loss percentage: 0
  Measurement: Round trip time
    Samples: 4, Minimum: 266 usec, Maximum: 388 usec, Average: 321 usec,
    Peak to peak: 122 usec, Stddev: 44 usec, Sum: 1283 usec
  Measurement: Positive round trip jitter
    Samples: 2, Minimum: 49 usec, Maximum: 74 usec, Average: 62 usec,
    Peak to peak: 25 usec, Stddev: 12 usec, Sum: 123 usec
  Measurement: Negative round trip jitter
    Samples: 1, Minimum: 50 usec, Maximum: 50 usec, Average: 50 usec,
    Peak to peak: 0 usec, Stddev: 0 usec, Sum: 50 usec
```

show services rpm probe-results

Syntax

```
show services rpm probe-results
<dst-interface interface-name>
<limit number>
<owner owner>
<source-address address>
<status (fail | pass) >
<target-address address>
<terse>
<test name>
```

Release Information

Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 on EX Series switches.
Command introduced in Junos OS Release 13.2 on PTX Series Packet Transport Series routers.
dst-interface, **limit**, **source-address**, **status**, **target-address**, and **terse** options introduced in Junos OS Release 18.1R1 on MX Series.
Command introduced in Junos OS Release 18.1 on QFX Series switches.

Description

Display the results of the most recent real-time performance monitoring (RPM) probes.

Options

All the following options require that you also configure the **owner** option.

dst-interface *interface-name*—(Optional) Display information only for RPM probes that are configured on this MS-MPC or MS-MIC services interface. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface.

limit *number*—(Optional) Limit the number of results that are displayed. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface.

Range: 1 through 4,294,967,295

Default: 100

none—Display information for all of the most recent RPM probes.

owner *owner*—(Optional) Display information only for probes with the specified probe owner. You must configure **owner** if you configure any other options.

source-address *address*—(Optional) Display information only for probes with the specified source address. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface.

status—(Optional) Display information only for probes with the specified type of test result. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface. Specify one of the following:

fail—Failed tests

pass—Passed tests

target-address *address*—(Optional) Display information only for probes with the specified target address. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface.

terse—(Optional) Display summary information. This option works only for RPM probes generated on an MS-MPC or MS-MIC services interface.

test *name*—(Optional) Display information only for the specified test.

Do not configure **test** if you configure any of the following options: **dst-interface**, **source-address**, or **target-address**. These options do not work when you configure **test**.

Required Privilege Level
view

List of Sample Output
show services rpm probe-results (IPv4 Targets) on page 1365
show services rpm probe-results (IPv6 Targets) on page 1368
show services rpm probe-results owner terse on page 1369
show services rpm probe-results owner status fail on page 1369
show services rpm probe-results (BGP Neighbor Discovery) on page 1369

Output Fields
Table 140 on page 1358 lists the output fields for the **show services rpm probe-results** command. Output fields are listed in the approximate order in which they appear.

Table 140: show services rpm probe-results Output Fields

Field Name	Field Description	Level of Output
Owner	Owner name. When you configure the probe owner statement at the **[edit services rpm]** hierarchy level, this field displays the configured owner name. When you configure BGP neighbor discovery through RPM, the output for this field is **Rpm-Bgp-Owner.**	none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Test	Name of a test representing a collection of probes. When you configure the test test-name statement at the **[edit services rpm probe owner]** hierarchy level, the field displays the configured test name. When you configure BGP neighbor discovery through RPM, the output for this field is **Rpm-BGP-Test-** *n*, where *n* is a cumulative number.	All levels
Target address	Destination IPv4 address used for the probes. This field is displayed when the probes are sent to the configured IPv4 or IPv6 targets or RPM servers.	none **dst-interface limit owner source-address target-address terse test**
Target inet6-address	Destination IPv6 address used for the probes. This field is displayed when the probes are sent to the configured IPv6 targets or RPM servers.	none **dst-interface limit owner source-address target-address terse test**
Source address	Source address used for the probes.	none **dst-interface limit owner source-address target-address test**
Probe type	Protocol configured on the receiving probe server: **http-get, http-metadata-get, icmp-ping, icmp-ping-timestamp, tcp-ping, udp-ping,** or **udp-ping-timestamp**.	none **dst-interface limit owner source-address target-address test**
Test size	Number of probes within a test.	none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Routing Instance Name	(BGP neighbor discovery) Name of the configured (if any) routing instance, logical system name, or both, in which the probe is configured: • When a routing instance is defined within a logical system, the logical system name is followed by the routing instance name. A slash **(/)** is used to separate the two entities. For example, if the routing instance called **R1** is configured within the logical system called **LS**, the name in the output field is **LS/R1**. • When a routing instance is configured but the default logical system is used, the name in the output field is the name of the routing instance. • When a logical system is configured but the default routing instance is used, the name in the output field is the name of the logical system followed by **default**. A slash (/) is used to separate the two entities. For example, **LS/default**.	none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output

Probe results	Raw measurement of a particular probe sample done by a remote host. This data is provided separately from the calculated results. The following information is contained in the raw measurement:
	• **Response received**
	  ◦ **Probe sent time**—Timestamp when the probe's results was sent.
	  ◦ **Probe rcvd/timeout time**—Timestamp when the probe's results was received.
	• **Client and server hardware timestamps**—If timestamps are configured, an entry appears at this point.
	• **Rtt**—Average ping round-trip time (RTT), in microseconds.
	• **Egress jitter**—Egress jitter, in microseconds.
	• **Ingress jitter**—Ingress jitter, in microseconds.
	• **Round trip jitter**—Round-trip jitter, in microseconds.
	• **Egress interarrival jitter**—Egress interarrival jitter, in microseconds.
	• **Ingress interarrival jitter**—Ingress interarrival jitter, in microseconds.
	• **Round trip interarrival jitter**—Round-trip interarrival jitter, in microseconds.
		none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Results over current test	Probes are grouped into tests, and the statistics are calculated for each test. If a test contains 10 probes, the average, minimum, and maximum results are calculated from the results of those 10 probes. If the command is issued while the test is in progress, the statistics use information from the completed probes. • **Probes sent**—Number of probes sent within the current test. • **Probes received**—Number of probe responses received within the current test. • **Loss percentage**—Percentage of lost probes for the current test. • **Measurement**—Measurement type. Possible values are round-trip time, positive round-trip jitter, negative round-trip jitter, egress time, positive egress jitter, negative egress jitter, ingress time, positive ingress jitter, negative ingress jitter, and, for the probe type **icmp-ping-timestamp**, the egress delay and ingress delay. For each measurement type, the following individual calculated results are provided: ◦ **Samples**—Number of probes. ◦ **Minimum**—Minimum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Maximum**—Maximum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Average**—Average RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Peak to peak**—Peak-to-peak difference, in microseconds. ◦ **Stddev**—Standard deviation, in microseconds. ◦ **Sum**—Statistical sum.	none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Results over last test	Results for the most recently completed test. If the command is issued while the first test is in progress, this information is not displayed • **Probes sent**—Number of probes sent for the most recently completed test. • **Probes received**—Number of probe responses received for the most recently completed test. • **Loss percentage**—Percentage of lost probes for the most recently completed test. • **Test completed**—Time the most recent test was completed. • **Measurement**—Measurement type. Possible values are round-trip time, positive round-trip jitter, negative round-trip jitter, egress time, positive egress jitter, negative egress jitter, ingress time, positive ingress jitter, negative ingress jitter, and, for the probe type **icmp-ping-timestamp**, the egress delay and ingress delay. For each measurement type, the following individual calculated results are provided: ◦ **Samples**—Number of probes. ◦ **Minimum**—Minimum RTT, ingress delay, or egress delay measured for the most recently completed test. ◦ **Maximum**—Maximum RTT, ingress delay, or egress delay measured for the most recently completed test. ◦ **Average**—Average RTT, ingress delay, or egress delay measured for the most recently completed test. ◦ **Peak to peak**—Peak-to-peak difference, in microseconds. ◦ **Stddev**—Standard deviation, in microseconds. ◦ **Sum**—Statistical sum.	none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Results over all tests	Displays statistics made for all the probes, independently of the grouping into tests, as well as statistics for the current test. • **Probes sent**—Number of probes sent in all tests. • **Probes received**—Number of probe responses received in all tests. • **Loss percentage**—Percentage of lost probes in all tests. • **Measurement**—Measurement type. Possible values are round-trip time, positive round-trip jitter, negative round-trip jitter, egress time, positive egress jitter, negative egress jitter, ingress time, positive ingress jitter, negative ingress jitter, and, for the probe types **icmp-ping-timestamp** and **udp-ping-timestamp**, the egress delay and ingress delay. For each measurement type, the following individual calculated results are provided: ◦ **Samples**—Number of probes. ◦ **Minimum**—Minimum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Maximum**—Maximum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Average**—Average RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Peak to peak**—Peak-to-peak difference, in microseconds. ◦ **Stddev**—Standard deviation, in microseconds. ◦ **Sum**—Statistical sum.	none **dst-interface limit owner source-address target-address test**

Table 140: show services rpm probe-results Output Fields *(continued)*

Field Name	Field Description	Level of Output
Error Stats	Displays error statistics for each probe. • **Invalid client recv timestamp**—Number of client receive timestamp less than client send timestamp. • **Invalid server send timestamp**—Number of server send timestamp less than server receive timestamp. • **Invalid server processing time**—Number of server side spent time greater than RTT. NOTE: **Error Stats** is displayed in the output only if non-zero statistics exists.	none **dst-interface limit owner source-address target-address test**
Last Probe Status	Status of the last probe that was sent for the current test (**fail** or **pass**).	**status**
Status	Status of the last completed test (**up** or **down**).	**status terse**
Source-IF	The MS-MPC or MS-MIC services interface that generates the RPM probes.	**terse**

Sample Output

show services rpm probe-results (IPv4 Targets)

`user@host>` **show services rpm probe-results**

```
Owner: ADSN-J4300.ADSN-J2300.D2, Test: 75300002
Target address: 172.16.54.172, Source address: 10.206.0.1,
Probe type: udp-ping-timestamp, Test size: 10 probes
Probe results:
  Response received
  Probe sent time: Tue Feb  6 14:53:15 2007,
  Probe rcvd/timeout time: Tue Feb 6 14:53:15 2007
  Client and server hardware timestamps
  Rtt: 575 usec, Egress jitter: 5 usec, Ingress jitter: 8 usec,
  Round trip jitter: 12 usec, Egress interarrival jitter: 8 usec,
 Ingress interarrival jitter: 7 usec, Round trip interarrival jitter: 7 usec,

  Round trip interarrival jitter: 669 usec
```

```
Results over current test:
  Probes sent: 10, Probes received: 10, Loss percentage: 0
  Measurement: Round trip time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec,  Sum: xxxx usec
  Measurement: Positive round trip jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative round trip jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
  Measurement: Egress time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
  Measurement: Positive Egress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative Egress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
  Measurement: Ingress time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
  Measurement: Positive Ingress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative Ingress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
Results over last test:
  Probes sent: 10, Probes received: 10, Loss percentage: 0
  Test completed on Tue Feb  6 14:53:16 2007
  Measurement: Round trip time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
  Measurement: Positive round trip jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative round trip jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
  Measurement: Egress time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
```

```
  Measurement: Positive Egress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative Egress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
  Measurement: Ingress time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
  Measurement: Positive Ingress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative Ingress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
Results over all tests:
  Probes sent: 560, Probes received: 560, Loss percentage: 0
  Measurement: Round trip time
    Samples: 560, Minimum: 805 usec, Maximum: 3114 usec, Average: 1756 usec,
    Peak to peak: 2309 usec, Stddev: 519 usec, Sum: xxxx usec
  Measurement: Positive round trip jitter
    Samples: 257, Minimum: 0 usec, Maximum: 2054 usec, Average: 597 usec,
    Peak to peak: 2054 usec, Stddev: 427 usec, Sum: xxxx usec
  Measurement: Negative round trip jitter
    Samples: 302, Minimum: 1 usec, Maximum: 1812 usec, Average: 511 usec,
    Peak to peak: 1811 usec, Stddev: 408 usec, Sum: xxxx usec
  Measurement: Egress time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
  Measurement: Positive Egress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative Egress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
  Measurement: Ingress time
    Samples: 10, Minimum: 805 usec, Maximum: 2859 usec, Average: 1644 usec,
    Peak to peak: 2054 usec, Stddev: 738 usec, Sum: xxxx usec
  Measurement: Positive Ingress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 2054 usec, Average: 876 usec,
    Peak to peak: 2049 usec, Stddev: 679 usec, Sum: xxxx usec
  Measurement: Negative Ingress  jitter
    Samples: 5, Minimum: 5 usec, Maximum: 1812 usec, Average: 926 usec,
    Peak to peak: 1807 usec, Stddev: 665 usec, Sum: xxxx usec
```

```
   Error Stats:
       Invalid client recv timestamp: 3, Invalid server send timestamp: 0
       Invalid server processing time: 0
```

show services rpm probe-results (IPv6 Targets)

`user@host>` **show services rpm probe-results**

```
   Owner: p, Test: t1
   Target inet6-address: 2001:db8:0:1:2a0:a502:0:1da,
   Target Port : 34567 Test size: 1000000 probes
   Probe results:
     Response received
     Probe sent time: Mon Dec 16 10:48:07 2013
     Probe rcvd/timeout time: Mon Dec 16 10:48:07 2013
     Client and server hardware timestamps
     Rtt: 236 usec, Round trip jitter: -10 usec, Round trip interarrival jitter:
484 usec
   Results over current test:
     Probes sent: 10, Probes received: 10, Loss percentage: 0
     Measurement: Round trip time
      Samples: 10, Minimum: 231 usec, Maximum: 298 usec, Average: 268 usec, Peak
to peak: 67 usec, Stddev: 24 usec, Sum: 2682 usec
     Measurement: Positive round trip jitter
       Samples: 3, Minimum: 15 usec, Maximum: 1841 usec, Average: 750 usec, Peak
to peak: 1826 usec, Stddev: 787 usec, Sum: 2251 usec
     Measurement: Negative round trip jitter
       Samples: 7, Minimum: 10 usec, Maximum: 1244 usec, Average: 709 usec, Peak
to peak: 1234 usec, Stddev: 466 usec, Sum: 4961 usec
   Results over last test:
     Probes sent: 10, Probes received: 10, Loss percentage: 0
     Test completed on Mon Dec 16 10:48:07 2013
     Measurement: Round trip time
      Samples: 10, Minimum: 231 usec, Maximum: 298 usec, Average: 268 usec, Peak
to peak: 67 usec, Stddev: 24 usec, Sum: 2682 usec
     Measurement: Positive round trip jitter
       Samples: 3, Minimum: 15 usec, Maximum: 1841 usec, Average: 750 usec, Peak
to peak: 1826 usec, Stddev: 787 usec, Sum: 2251 usec
     Measurement: Negative round trip jitter
       Samples: 7, Minimum: 10 usec, Maximum: 1244 usec, Average: 709 usec, Peak
to peak: 1234 usec, Stddev: 466 usec, Sum: 4961 usec
   Results over all tests(From start of current control session):
     Probes sent: 490, Probes received: 488, Loss percentage: 0
     Measurement: Round trip time
```

```
        Samples: 488, Minimum: 231 usec, Maximum: 306 usec, Average: 270 usec,
Peak to peak: 75 usec, Stddev: 16 usec, Sum: 131586 usec
      Measurement: Positive round trip jitter
        Samples: 254, Minimum: 0 usec, Maximum: 10151 usec, Average: 157 usec,
Peak to peak: 10151 usec, Stddev: 873 usec, Sum: 39817 usec
      Measurement: Negative round trip jitter
        Samples: 233, Minimum: 1 usec, Maximum: 10170 usec, Average: 171 usec,
Peak to peak: 10169 usec, Stddev: 888 usec, Sum: 39889 usec
```

show services rpm probe-results owner terse

```
user@host> show services rpm probe-results owner owner1 terse
```

```
Test Name      Source-IF           Target Address      Status       Last Change
t_1            ms-2/2/0.10         192.0.2.1           UP           0D0H1M29S

t_2            ms-2/2/0.10         192.0.2.1           UP           0D0H1M29S

t_3            ms-2/2/0.10         192.0.2.1           UP           0D0H1M29S
```

show services rpm probe-results owner status fail

```
user@host> show services rpm probe-results owner owner1 status fail
```

```
Test Name              Last Probe Status         Status
t_1                    FAIL                      DOWN
t_2                    FAIL                      DOWN
t_3                    FAIL                      DOWN
```

show services rpm probe-results (BGP Neighbor Discovery)

```
user@host> show services rpm probe-results
```

```
Owner: Rpm-Bgp-Owner, Test: Rpm-Bgp-Test-1
    Target address: 10.209.152.37, Probe type: icmp-ping, Test size: 5 probes
    Routing Instance Name: LS1/RI1
    Probe results:
      Response received
      Probe sent time: Fri Oct 28 05:20:23 2005
      Probe rcvd/timeout time: Fri Oct 28 05:20:23 2005
      Rtt: 662 usec
    Results over current test:
```

```
  Probes sent: 5, Probes received: 5, Loss percentage: 0
  Measurement: Round trip time
    Minimum: 529 usec, Maximum: 662 usec, Average: 585 usec,
    Jitter: 133 usec, Stddev: 53 usec
Results over all tests:
  Probes sent: 5, Probes received: 5, Loss percentage: 0
  Measurement: Round trip time
    Minimum: 529 usec, Maximum: 662 usec, Average: 585 usec,
    Jitter: 133 usec, Stddev: 53 usec
```

show services rpm rfc2544-benchmarking

Syntax

```
show services rpm rfc2544-benchmarking
<aborted-tests (test-id [test-id] | brief | detail)>
<active-tests (test-id [test-id] | brief | detail)>
<completed-tests (test-id [test-id] | brief | detail)>
<summary>
```

Release Information

Command introduced in Junos OS Release 12.3X52 for ACX Series routers.
Command introduced in Junos OS Release 13.3R1 for MX104 Universal Routing Platforms.

Description

Display information about the results of each category or state of the RFC 2544-based benchmarking test, such as aborted tests, active tests, and completed tests, for each real-time performance monitoring (RPM) instance. You can view the results of each test state for all of the configured test IDs or for a specific test ID. Also, you can display statistics about the total number of tests of each state for a high-level, quick analysis. The values in the output displayed vary, depending on the state in which the test is passing through, when you issue the command.

You can view the test results of multiple test IDs at the same time by entering the IDs in a single command. If you enter multiple test ID values, you must separate each number with a space.

Options

none—Display test results for all categories.

aborted-tests—(Optional) Display the list of tests that were aborted or stopped. This list includes tests that failed due to various error conditions and tests that you terminated by entering the **test service rpm rfc2544-benchmarking test *test-name* stop** command. The **Status** field in the output specifies the reason for the termination of the test.

active-tests—(Optional) Display the results of the set of tests that are currently running.

brief | detail—Display the specified level of output.

completed-tests—(Optional) Display the results of the set of tests that were successfully completed. A completed test is one that passes through all the test steps or states specified in RFC 2544. A test that is marked as completed after it went through all the states from the beginning to the end can still be reported as a failed test. For example, a failed test can be a test that sends the desired number of packets, but does not receive the frames back from the other end.

summary—(Optional) Display summary output.

test-id *test-id*—Display test results for the specified unique identifier of the test.

Required Privilege Level
view

RELATED DOCUMENTATION

- Configuring an RFC 2544-Based Benchmarking Test | **659**
- Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**
- rfc2544-benchmarking | **1108**

List of Sample Output

Output Fields

Table 141 on page 1372 lists the output fields for the **show services rpm rfc2544-benchmarking** command. Output fields are listed in the approximate order in which they appear.

Table 141: show services rpm rfc2544-benchmarking Output Fields

Field Name	Field Description
Test information	Details of the performed RFC 2544 benchmarking test.
Test id	Unique identifier configured for the test.
Test name	Name configured for the test.
Test type	The type of statistical detail that is collected for the test, based on the configured test type. Throughput-related, latency, frame-loss, or back-to-back frames-related information is displayed for ACX Series routers. Reflected packets-related information is displayed for MX104 Series routers.

Table 141: show services rpm rfc2544-benchmarking Output Fields *(continued)*

Field Name	Field Description
Test mode	Mode configured for the test on the router. Test modes are: • Initiate-and-Terminate: Test frames are initiated from one end and terminated at the same end. This mode requires a reflector to be configured at the peer end to enable the test frames to be returned to the source. This mode is supported only on ACX Series routers. • Reflect: Test frames that originate from one end are reflected at the other end on the selected service, such as IPv4 or Ethernet.
Test packet size	Size of the test packets in bytes. This field is valid only when the test mode is Initiate-and-Terminate.
Test state	State of the test that is in progress or active when the output is displayed.
Status	Indicates whether the test is currently in progress or has been terminated. This field is displayed for tests that are in progress or were aborted by entering the **test services rpm rfc2544-benchmarking test <*test-name* | *test-id*> stop** command.
Test start time	Time at which the test started in Coordinated Universal Time (UTC) format (YYYY-MM-DD-HH:MM:SS).
Test finish time	Time at which the test completed.
Counters last cleared	Date, time, and how long ago the statistics for the test were cleared. The format is *year-month-day hour:minute:second:timezone* (*hour:minute:second* ago). For example, 2010-05-17 07:51:28 PDT (00:04:33 ago). If you did not clear the statistics previously at any point, **Never** is displayed.
Number of active tests	Total number of tests that are currently running.
Number of completed tests	Total number of tests that were successfully completed
Number of aborted tests	Total number of tests that were aborted or halted.

Sample Output

show services rpm rfc2544-benchmarking summary

`user@host>` **show services rpm rfc2544-benchmarking summary**

```
Rfc2544 tests summary :
      Number of active tests: 0, Number of completed tests: 4, Number of aborted
tests: 52
```

This output indicates that no test iteration is currently in progress (at the time of issue of the command), 4 tests were completed successfully, and 52 tests were halted.

show services rpm rfc2544-benchmarking aborted-tests (ACX Series Router)

`user@host>` **show services rpm rfc2544-benchmarking aborted-tests**

```
Test information :
      Test id: 1, Test name: test1, Test type: Throughput
      Test mode: Initiate-and-Terminate
      Test packet size: 64 1280
      Test state: RFC2544_TEST_STATE_STOPPED
      Status: User-aborted-via-cli
      Test start time: 2005-08-05 03:19:58 UTC
      Test finish time: 2005-08-05 03:20:00 UTC
      Counters last cleared: Never

      Test id: 2, Test name: test1, Test type: Throughput
      Test mode: Initiate-and-Terminate
      Test packet size: 64 1280
      Test state: RFC2544_TEST_STATE_STOPPED
      Status: User-aborted-via-cli
      Test start time: 2005-08-05 03:20:00 UTC
      Test finish time: 2005-08-05 03:20:02 UTC
      Counters last cleared: Never
```

show services rpm rfc2544-benchmarking completed-tests (ACX Series Router)

`user@host>` **show services rpm rfc2544-benchmarking completed-tests**

```
 Test information :
      Test id: 18, Test name: test1, Test type: Throughput
      Test mode: Initiate-and-Terminate
```

```
      Test packet size: 64 1280
      Test state: RFC2544_TEST_STATE_COMPLETED
      Test start time: 2005-08-05 03:20:34 UTC
      Test finish time: 2005-08-05 03:21:23 UTC
      Counters last cleared: Never
```

show services rpm rfc2544-benchmarking active-tests (ACX Series Router)

`user@host>` **show services rpm rfc2544-benchmarking active-tests**

```
Test information :
      Test id: 57, Test name: test1, Test type: Back-Back-Frames
      Test mode: Initiate-and-Terminate
      Test packet size: 64 1280
      Test state: RFC2544_TEST_STATE_RUNNING
      Status: Running
      Test start time: 2005-08-05 20:15:41 UTC
      Test finish time: TEST_RUNNING
      Counters last cleared: Never
```

show services rpm rfc2544-benchmarking aborted-tests (MX104 Router)

`user@host>` **show services rpm rfc2544-benchmarking aborted-tests**

```
  Test information :
      Test id: 1, Test name: prof_tput1, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: TEST_STATE_STOPPED
      Status: Test-intf-ifl-change
      Test start time: 2013-12-16 22:54:27 PST
      Test finish time: 2013-12-16 23:30:28 PST
      Counters last cleared: Never

      Test id: 2, Test name: prof_tput1, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: TEST_STATE_STOPPED
      Status: User-aborted-via-cli
      Test start time: 2013-12-16 23:31:06 PST
      Test finish time: 2013-12-16 23:36:22 PST
      Counters last cleared: Never
```

```
     Test id: 3, Test name: prof_tput1, Test type: Reflect
     Test mode: Reflect
     Test packet size: 0
     Test state: TEST_STATE_STOPPED
     Status: User-aborted-via-cli
     Test start time: 2013-12-16 23:36:24 PST
     Test finish time: 2013-12-17 01:49:24 PST
     Counters last cleared: Never
```

show services rpm rfc2544-benchmarking completed-tests (MX104 Router)

user@host> **show services rpm rfc2544-benchmarking completed-tests**

```
Test information :
     Test id: 18, Test name: test1, Test type: Reflect
     Test mode: Reflect
     Test packet size: 0
     Test state: TEST_STATE_COMPLETED
     Test start time: 2005-08-05 03:20:34 UTC
     Test finish time: 2005-08-05 03:21:23 UTC
     Counters last cleared: Never
```

show services rpm rfc2544-benchmarking active-tests (MX104 Router)

user@host> **show services rpm rfc2544-benchmarking active-tests**

```
 Test information :
     Test id: 4, Test name: prof_tput1, Test type: Reflect
     Test mode: Reflect
     Test packet size: 0
     Test state: TEST_STATE_RUNNING
     Status: Running
     Test start time: 2013-12-17 01:49:26 PST
     Test finish time: TEST_RUNNING
     Counters last cleared: Never
```

show services rpm rfc2544-benchmarking test-id

Syntax

```
show services rpm rfc2544-benchmarking test-id test-id
<brief | detail>
```

Release Information

Command introduced in Junos OS Release 12.3X52 for ACX Series routers.
Command introduced in Junos OS Release 13.3R1 for MX104 Universal Routing Platforms.

Description

Display information about the results of the RFC 2544-based benchmarking test for a specific test ID for each real-time performance monitoring (RPM) instance. The values in the output displayed vary, depending on the state in which the test is passing through, when you issue the command.

Options

none—Display brief information about a specific test ID of the benchmarking test.

test-id *test-id* —Display test results for the specified unique identifier.

brief | detail—(Optional) Display the specified level of output.

Required Privilege Level

view

RELATED DOCUMENTATION

Configuring an RFC 2544-Based Benchmarking Test | 659

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | 647

rfc2544-benchmarking | 1108

List of Sample Output

show services rpm rfc2544-benchmarking test-id detail (Reflection Test on MX104 Routers) on page 1397
show services rpm rfc2544-benchmarking test-id brief (Reflection Test on MX104 Routers) on page 1398

Output Fields

Table 142 on page 1378 lists the output fields for the **show services rpm rfc2544-benchmarking test-id** command. Output fields are listed in the approximate order in which they appear.

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields

Field Name	Field Description	Level of Output
Test information	Details of the performed RFC 2544 benchmarking test.	None specified
Test id	Unique identifier configured for the test.	None specified
Test name	Name configured for the test.	None specified
Test type	The type of actual test run that is collected for the test, based on the configured test type. Throughput-related, latency, frame-loss, or back-to-back frames-related information is displayed for ACX Series routers. Reflected packets-related information is displayed for MX104 Series routers.	None specified
Test mode	Mode configured for the test on the router. Test modes are: • Initiate-and-Terminate: Test frames are initiated from one end and terminated at the same end. This mode requires a reflector to be configured at the peer end to enable the test frames to be returned to the source. This mode is supported only on ACX Series routers. • Reflect: Test frames that originate from one end are reflected back to the originator, such as IPv4 or Ethernet.	None specified
Test packet size	Size of the test packets in bytes. This field is valid only when the test mode is Initiate-and-Terminate.	None specified
Test state	State of the test that is in progress or active when the output is displayed. For details about the states, see *RFC 2544-Based Benchmarking Test States*.	None specified
Status	Indicates whether the test is currently in progress or has been terminated.	None specified
Test start time	Time at which the test started in Coordinated Universal Time (UTC) format (YYYY-MM-DD-HH:MM:SS).	None specified
Test finish time	Time at which the test completed.	None specified

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Counters last cleared	Date, time, and how long ago the statistics for the test were cleared. The format is *year-month-day hour:minute:second:timezone* (*hour:minute:second* ago). For example, 2010-05-17 07:51:28 PDT (00:04:33 ago). If you did not clear the statistics previously at any point, **Never** is displayed.	None specified
Test-profile Configuration	(ACX Series routers only) Details of the specified test profile	**detail**
Test-profile name	(ACX Series routers only) Name of the configured test profile that contains the parameters for the test	**detail**
Test packet size	(ACX Series routers only) Size of the test packets in bytes	**detail**
Theoretical max bandwidth	(ACX Series routers only) Theoretical maximum bandwidth configured for the test. This value is typically set to the bandwidth of the server being tested. Valid values are 1 Kbps through 1,000,000 Kbps (1 Gbps). The value defined is the highest bandwidth value tested for this test.	**detail**
Test Configuration	Details of the configured test ID.	**detail**
Test mode	Mode configured for the test. Test modes are Initiate-and-Terminate and Reflect.	**detail**
Duration in seconds	Period in seconds for which the test has been performed.	**detail**
Test family	The underlying service on which the test is run. Test families are: • INET: Indicates that the test is run on a IPV4 service. • CCC: Indicates that the test is run on a circuit cross-connect (CCC) or pseudowire service.	**detail**
Routing Instance Name	(ACX Series routers only) Name of the routing instance for the test	**detail**
Inet family Configuration	Details of the configured **inet** family for an IPv4 service	**detail**
Egress Interface	Name of the egress interface from which the test frames are sent	**detail**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Source ipv4 address	Source IPv4 address used in the IP header of the generated test frame.	**detail**
Destination ipv4 address	Destination IPv4 address used in the IP header of the generated test frame.	**detail**
Source udp port	Source UDP port number used in the UDP header of the generated test frame.	**detail**
Destination udp port	Destination UDP port number used in the UDP header of the generated test frame.	**detail**
Ccc family Configuration	Details of the configured CCC family for an Ethernet service	**detail**
Source MAC address	(ACX Series routers only) Source MAC address used in generated test frames for a CCC or Ethernet pseudowire service.	**detail**
Destination MAC address	(ACX Series routers only) Destination MAC address used in generated test frames for a CCC or Ethernet pseudowire service.	**detail**
Ivlan-id	(ACX Series routers only) Inner VLAN ID for test-frames.	**detail**
Ovlan-id	(ACX Series routers only) Outer VLAN ID for test-frames.	**detail**
Direction egress	Test is run in the egress direction of the interface (NNI)	**detail**
Direction ingress	Test is run in the ingress direction of the interface (UNI)	**detail**
Rfc2544 throughput test information	**(ACX Series routers only) Details of the throughput test**	**detail**
Initial test load percentage	Percentage of the steady state load for the test.	**detail**
Test iteration mode	Mode of the test iteration: Binary or step-down.	**detail**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Test iteration step percent	The test step percentage for tests. If not specified, the default step-percent is 10 percent. This parameter is ignored for all type of tests other than frame-loss tests.	**detail**
Theoretical max bandwidth	The theoretical limit of the media for the frame size configured for the test. This value is typically set to the bandwidth of the server being tested.	**detail**
Test packet size:	Packet size of the test frames in bytes.	**detail**
Iteration	Number of the test iteration.	**detail**
Duration (sec)	Period in seconds for which the test iteration is run	**detail**
Elapsed time	Amount of time that has passed, in seconds, since the start of the test.	**detail**
pps	Total count of packets-per-second (pps) transmitted during the test.	**detail**
Tx Packets	Number of transmitted test packets.	**detail**
Rx Packets	Number of received test packets.	**detail**
Tx Bytes	Number of transmitted bytes.	**detail**
Rx Bytes	Number of received bytes.	**detail**
Percentage throughput	Percentage of throughput for the test iteration.	**detail**
Result of the iteration runs (Throughput) :	Results of the completed throughput test for a particular packet size.	**detail**
Best iteration	Number of the iteration with the highest throughout, among the listed iterations.	**detail**
Best iteration (pps)	Packets-per-second (pps) count of the iteration with the highest throughout, among the listed iterations.	**detail**
Best iteration throughput	Percentage of throughput of the iteration with the highest throughout, among the listed iterations.	**detail**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
RFC2544 Throughput test results summary	Consolidated information of the throughput test.	**detail summary**
Packet Size	Size of the test packet in bytes.	**detail summary**
Theoretical rate (pps)	Theoretical frame rate in packets-per-second.	**detail summary**
Tx Packets	Number of transmitted packets.	**detail summary**
Rx Packets	Number of received packets.	**detail summary**
Offered throughput (percentage)	The offered throughput in percentage of the chosen service (such as Layer 3 or Ethernet pseudowire).	**detail summary**
Measured bandwidth (kbps)	Available bandwidth of the service based on the calculated throughput.	**detail summary**
Rfc2544 latency test information :	**(ACX Series routers only) Details of the latency test**	**detail**
Theoretical max bandwidth	Theoretical maximum bandwidth configured for the test. This value is typically set to the bandwidth of the server being tested. Valid values are 1 Kbps through 1,000,000 Kbps (1 Gbps). The value defined is the highest bandwidth value used for this test.	**detail**
Initial test load percentage	Percentage of the steady state load for the test.	**detail**
Duration in seconds	Period in seconds for which the test has been performed.	**detail**
Test packet size	Size of the test packet in bytes.	**detail**
Iteration	Number of the test iteration.	**detail**
Duration (sec)	Period in seconds for which the test iteration is run.	**detail**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Elapsed time	Amount of time that has passed, in seconds, since the start of the test.	**detail**
pps	Total count of packets-per-second (pps) transmitted during the test.	**detail**
Tx Packets	Number of transmitted test packets.	**detail**
Rx Packets	Number of received test packets.	**detail**
Latency	Displays the latency parameters.	**detail**
Min(ns)	Aggregated minimum latency in nanoseconds.	**detail**
Avg(ns)	Aggregated average latency in nanoseconds.	**detail**
Max(ns)	Aggregated maximum latency in nanoseconds.	**detail**
Probe(ns)	Aggregated probe latency in nanoseconds.	**detail**
Result of the iteration runs (Latency)	Results of the latency test completed for a particular packet size.	**detail**
Avg (min) Latency	Average of the minimum latency in nanoseconds.	**detail**
Avg (avg) latency	Average of the average latency in nanoseconds.	**detail**
Avg (Max) latency	Average of the maximum latency in nanoseconds.	**detail**
Avg (probe) latency	Average of the probe latency in nanoseconds.	**detail**
RFC2544 Latency test results summary:	Consolidated statistics of the latency test.	**detail summary**
Packet Size	Size of the test packet in bytes.	**detail summary**
Theoretical rate (pps)	Theoretical frame rate in packets-per-second.	**detail summary**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Tx Packets	Number of transmitted packets.	**detail summary**
Rx Packets	Number of received packets.	**detail summary**
Latency	Displays the latency parameters.	**detail summary**
Min(ns)	Aggregated minimum latency in nanoseconds.	**detail summary**
Avg(ns)	Aggregated average latency in nanoseconds.	**detail summary**
Max(ns)	Aggregated maximum latency in nanoseconds.	**detail summary**
Probe(ns)	Aggregated probe latency in nanoseconds.	**detail summary**
Rfc2544 Back-Back test information :	**(ACX Series routers only) Details of the back-to-back frames or bursty frames test.**	**detail**
Initial burst length:	Length of the first burst when test frames are sent, as a measure of number of seconds at the rate of Kbps.	**detail**
Test iteration mode :	Mode of the test iteration: Binary or step-down.	**detail**
Test iteration step percent	The test step percentage for tests. If not specified, the default step-percent is 10 percent. This parameter is ignored for all type of tests other than frame-loss tests.	**detail**
Theoretical max bandwidth	The theoretical limit of the media for the frame size configured for the test. This value is typically set to the bandwidth of the server being tested.	**detail**
Test packet size:	Packet size of the test frames in bytes.	**detail**
Iteration	Number of the test iteration.	**detail**
Burst Length (Packets)	Number of packets in the burst.	**detail**
Elapsed time	Amount of time that has passed, in seconds, since the start of the test.	**detail**
Tx Packets	Number of transmitted test packets.	**detail**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Rx Packets	Number of received test packets.	**detail**
Tx Bytes	Number of transmitted bytes.	**detail**
Rx Bytes	Number of received bytes.	**detail**
Result of the iteration runs :	Results of the back-to-back frames test completed for a certain packet size.	**detail**
Best iteration :	Number of the iteration with the longest burst.	**detail**
Measured burst (num sec)	Time in seconds of the burst of the iteration with the longest burst.	**detail**
Measured burst (num pkts)	Number of packets during the burst of the iteration with the longest burst.	**detail**
RFC2544 Back-Back test results summary:	Consolidated statistics of the back-to-back frames test.	**detail summary**
Packet Size	Size of the test packets in bytes.	**detail summary**
Measure Burst length (Packets)	Computed burst length in terms of number of packets.	**detail summary**
Rfc2544 frame-loss test information :	**(ACX Series routers only) Details of the frame-loss test.**	**detail**
Initial burst length:	Length of the first burst when test frames are sent, as a measure of number of seconds at the rate of Kbps.	**detail**
Test iteration mode :	Mode of the test iteration: Binary or step-down.	**detail**
Test iteration step percent	The test step percentage for tests. If not specified, the default step-percent is 10 percent. This parameter is ignored for all type of tests other than frame-loss tests.	**detail**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Theoretical max bandwidth	The theoretical limit of the media for the frame size configured for the test. This value is typically set to the bandwidth of the server being tested.	**detail**
Test packet size	Size of the test packets in bytes.	**detail**
Iteration	Number of the test iteration.	**detail**
Duration (sec)	Period, in seconds, for which the test iteration is run.	**detail**
Offered throughput (percentage)	The offered throughput in percentage of the chosen service (such as Layer 3 or Ethernet pseudowire)	**detail**
Elapsed time	Amount of time that has passed, in seconds, since the start of the test.	**detail**
pps	Theoretical frame rate in packets-per-second.	**detail**
Tx Packets	Number of transmitted test packets.	**detail**
Rx Packets	Number of received test packets.	**detail**
Tx Bytes	Number of transmitted bytes.	**detail**
Rx Bytes	Number of received bytes.	**detail**
Frame-loss rate %	Percentage of frames that must been forwarded by the router under steady state (constant) load, but were not forwarded due to lack of resources.	**detail**
Result of the iteration runs :	Results of the frame-loss test completed for a certain packet size.	**detail**
Frame-loss rate (percent) :	Percentage of dropped frames for the specified packet size	**detail**
RFC2544 Frame-loss test results summary	Consolidated statistics of the frame-loss test	**detail**
Packet Size	Size of the test packet in bytes.	**detail summary**

Table 142: show services rpm rfc2544-benchmarking test-id Output Fields *(continued)*

Field Name	Field Description	Level of Output
Theoretical rate (pps)	Theoretical frame rate in packets-per-second.	**detail summary**
Percentage throughput	Percentage of throughput for the test iteration.	**detail summary**
Tx Packets	Number of transmitted packets.	**detail summary**
Rx Packets	Number of received packets.	**detail summary**
Frame Loss rate percent	Percentage of dropped frames for the specified packet size	**detail summary**

Sample Output

show services rpm rfc2544-benchmarking test-id detail (Throughput Test on ACX Series Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id 19 detail**

```
Test information :
      Test id: 19, Test name: test1, Test type: Throughput
      Test mode: Initiate-and-Terminate
      Test packet size: 64 1280
      Test state: RFC2544_TEST_STATE_COMPLETED
      Test start time: 2005-07-29 10:25:00 UTC
      Test finish time: 2005-07-29 10:26:02 UTC
      Counters last cleared: Never

      Test-profile Configuration:
          Test-profile name: prof_tput
          Test packet size: 64 1280
          Therotical max bandwidth : 993000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 20
          Test family: INET
          Routing Instance Name: default
```

```
      Inet family Configuration:
          Egress Interface : ge-0/1/1.0
          Source ipv4 address: 192.0.2.1
          Destination ipv4 address: 192.0.2.2
          Source udp port: 2020
          Destination udp port: 3030

  Rfc2544 throughput test information :
      Initial test load percentage : 100.00 %
      Test iteration mode : Binary
      Test iteration step percent : 50.00 %
      Therotical max bandwidth : 993000 kbps

Test packet size: 64
Iteration Duration Elapsed pps      Tx       Rx       Tx         Rx
Percentage
           (sec)    time            Packets  Packets  Bytes      Bytes
throughput
   1       3        3      134918   404754   404754   27523272   27523272   10.00
 %
   2       20       20     1349184  26983501 26983501 1834878068 1834878068 100.00
 %

Result of the iteration runs : Throughput Test complete for packet size 64
Best iteration : 2, Best iteration (pps) : 1349184
Best iteration throughput : 100.00 %

Test packet size: 1280
Iteration Duration Elapsed pps    Tx      Rx      Tx        Rx        Percentage

          (sec)    time         Packets  Packets  Bytes     Bytes     throughput

   1       3        3      9489   28467    28467    36551628   36551628   10.00 %

   2       20       20     94896  1897920  1897920  2436929280 2436929280 100.00
%

Result of the iteration runs : Throughput Test complete for packet size 1280
Best iteration : 2, Best iteration (pps) : 94896
Best iteration throughput : 100.00 %

RFC2544 Throughput test results summary:
----------------------------------------
```

```
  Packet  Theoretical   Tx         Rx        Offered throughput   Measured
   Size   rate (pps)    Packets    Packets       (percentage)     bandwidth (kbps)

   64     1349184       26983501   26983501        100.00 %       993000
   1280   94896         1897920    1897920         100.00 %       993000
```

show services rpm rfc2544-benchmarking test-id detail (Latency Test on ACX Series Routers)

user@host> **show services rpm rfc2544-benchmarking test-id 37 detail**

```
  Test information :
      Test id: 37, Test name: test1, Test type: Latency
      Test mode: Initiate-and-Terminate
      Test packet size: 64 1280
      Test state: RFC2544_TEST_STATE_COMPLETED
      Test start time: 2005-07-29 10:26:41 UTC
      Test finish time: 2005-07-29 10:36:15 UTC
      Counters last cleared: Never

      Test-profile Configuration:
          Test-profile name: prof_latency
          Test packet size: 64 1280
          Therotical max bandwidth : 993000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 10
          Test family: INET
          Routing Instance Name: default

      Inet family Configuration:
          Egress Interface : ge-0/1/1.0
          Source ipv4 address: 192.0.2.1
          Destination ipv4 address: 192.0.2.2
          Source udp port: 2020
          Destination udp port: 3030

  Rfc2544 latency test information :
      Therotical max bandwidth : 993000 kbps
      Initial test load percentage : 100.00 %
      Duration in seconds: 10

Test packet size: 64
```

```
Iteration  Duration   Elapsed     pps            Tx           Rx
            (sec)      time                   Packets      Packets
     1           3         3       134918       404754       404754
     2          10        10      1349184     13491751     13491751
     3          10        10      1349184     13491751     13491751
     4          10        10      1349184     13491751     13491751
     5          10        10      1349184     13491751     13491751
     6          10        10      1349184     13491751     13491751
     7          10        10      1349184     13491751     13491751
     8          10        10      1349184     13491751     13491751
     9          10        10      1349184     13491751     13491751
    10          10        10      1349184     13491751     13491751
    11          10        10      1349184     13491751     13491751
    12          10        10      1349184     13491751     13491751
    13          10        10      1349184     13491751     13491751
    14          10        10      1349184     13491751     13491751
    15          10        10      1349184     13491751     13491751
    16          10        10      1349184     13491751     13491751
    17          10        10      1349184     13491751     13491751
    18          10        10      1349184     13491751     13491751
    19          10        10      1349184     13491751     13491751
    20          10        10      1349184     13491751     13491751
    21          10        10      1349184     13491751     13491751

-------------- Latency ---------------
Min(ns)    Avg(ns)    Max(ns)    Probe(ns)
17464       18770      18880        18784
17472       18799      20488        18848
17472       18799      20416        18816
17472       18799      20440        18704
17464       18799      20376        18880
17464       18799      20232        18832
17464       18799      20400        18848
17472       18799      20240        18864
17472       18799      20264        18848
17464       18799      20264        18880
17472       18800      20320        18864
17464       18799      20176        18864
17464       18800      20248        18864
17464       18800      20272        18864
17464       18799      20472        18832
17464       18799      20256        18880
17464       18799      20336        18848
```

```
17464       18800      20688        18848
17472       18800      20504        18864
17464       18799      20448        18768
17472       18799      20240        18864

Result of the iteration runs : Latency Test complete for packet size 64
Avg (min) Latency                  : 17466
Avg (avg) latency                  : 18799
Avg (Max) latency                  : 20360
Avg (probe) latency                : 18844

Test packet size: 1280

Iteration  Duration   Elapsed     pps            Tx           Rx
            (sec)      time                   Packets      Packets
     1           3         3         9489        28467        28467
     2          10        10        94896       948960       948960
     3          10        10        94896       948960       948960
     4          10        10        94896       948960       948960
     5          10        10        94896       948960       948960
     6          10        10        94896       948960       948960
     7          10        10        94896       948960       948960
     8          10        10        94896       948960       948960
     9          10        10        94896       948960       948960
    10          10        10        94896       948960       948960
    11          10        10        94896       948960       948960
    12          10        10        94896       948960       948960
    13          10        10        94896       948960       948960
    14          10        10        94896       948960       948960
    15          10        10        94896       948960       948960
    16          10        10        94896       948960       948960
    17          10        10        94896       948960       948960
    18          10        10        94896       948960       948960
    19          10        10        94896       948960       948960
    20          10        10        94896       948960       948960
    21          10        10        94896       948960       948960

-------------- Latency ---------------
Min(ns)    Avg(ns)    Max(ns)    Probe(ns)
68712       70031      70576        69456
68728       70344      71808        70512
68720       70344      71744        70352
68720       70344      71680        70112
68720       70345      71856        70352
```

```
68720       70344      71808        70384
68720       70344      71752        70480
68720       70344      71880        70112
68720       70344      71792        70320
68728       70345      73344        70336
68720       70344      71688        70560
68728       70345      71896        70496
68720       70344      71760        70096
68720       70344      71776        70320
68720       70344      71760        70400
68712       70345      71920        70352
68720       70344      71792        70576
68720       70345      71840        70320
68720       70344      71792        70368
68720       70345      71824        70464
68712       70345      71904        70512


Result of the iteration runs : Latency Test complete for packet size 1280
Avg (min) Latency                  : 68720
Avg (avg) latency                  : 70344
Avg (Max) latency                  : 71880
Avg (probe) latency                : 70371

RFC2544 Latency test results summary:
---------------------------------------

  Packet  Theoretical Tx        Rx        ------------- Latency -------------
   Size   rate (pps)  Packets   Packets   Min(ns)  Avg(ns)  Max(ns) Probe(ns)
   64     1349184     269835020 269835020 17466    18799    20360    18844
   1280   94896       18979200  18979200  68720    70344    71880    70371
```

show services rpm rfc2544-benchmarking test-id detail (Frame Loss Test on ACX Series Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id 73 detail**

```
  Test information :
       Test id: 73, Test name: test1, Test type: Frame-Loss
       Test mode: Initiate-and-Terminate
       Test packet size: 64 1280
       Test state: RFC2544_TEST_STATE_COMPLETED
       Test start time: 2005-07-29 10:38:41 UTC
       Test finish time: 2005-07-29 10:41:19 UTC
```

```
       Counters last cleared: Never

       Test-profile Configuration:
           Test-profile name: prof_fl
           Test packet size: 64 1280
           Therotical max bandwidth : 993000 kbps

       Test Configuration:
           Test mode: Initiate-and-Terminate
           Duration in seconds: 20
           Test family: INET
           Routing Instance Name: default

       Inet family Configuration:
           Egress Interface : ge-0/1/1.0
           Source ipv4 address: 192.0.2.1
           Destination ipv4 address: 192.0.2.2
           Source udp port: 2020
           Destination udp port: 3030

   Rfc2544 frame-loss test information :
       Initial test load percentage : 100.00 %
       Test iteration mode : step-down
       Test iteration step percent : 10 %
       Therotical max bandwidth : 993000 kbps

 Test packet size: 64
 Iteration Duration  Elapsed  Offered     pps    Tx       Rx       Tx         Rx
       Frame-loss
             (sec)  time    throughput%         Packets  Packets  Bytes      Bytes
      rate %
  1       3         3       10.00 %     134918  404754   404754   27523272
27523272   0.00 %
  2       20        20     100.00 %     1349184 26983501 26983501 1834878068
1834878068 0.00 %
  3       20        20     100.00 %     1349184 26983501 26983501 1834878068
1834878068 0.00 %
  4       20        20     100.00 %     1349184 26983501 26983501 1834878068
1834878068 0.00 %


 Result of the iteration runs : Frame-loss test complete for packet size 64
 Frame-loss rate (percent) : 0.00 %
```

```
 Test packet size: 1280
 Iteration Duration  Elapsed  Offered     pps    Tx       Rx       Tx         Rx
       Frame-loss
             (sec)  time    throughput%         Packets  Packets  Bytes      Bytes
      rate %
  1       3         3       10.00 %     9489    404754   28467    36551628
36551628   0.00 %
  2       20        20     100.00 %     94896   1897920  1897920  2436929280
2436929280 0.00 %
  3       20        20     100.00 %     94896   1897920  1897920  2436929280
2436929280 0.00 %
  4       20        20     100.00 %     94896   1897920  1897920  2436929280
2436929280 0.00 %


 Result of the iteration runs : Frame-loss test complete for packet size 1280
 Frame-loss rate (percent) : 0.00 %

 RFC2544 Frame-loss test results summary:
 ----------------------------------------

   Packet      Theoretical     Percentage      Tx             Rx           Frame
Loss
    Size       rate (pps)      throughput    Packets        Packets       rate
percent
       64         1349184       100.00 %      26983501       26983501        0.00
 %
     1280           94896       100.00 %       1897920        1897920        0.00
 %
```

show services rpm rfc2544-benchmarking test-id detail (Back-to-Back Frames Test on ACX Series Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id 55 detail**

```
Test information :
       Test id: 55, Test name: test1, Test type: Back-Back-Frames
       Test mode: Initiate-and-Terminate
       Test packet size: 64 1280
       Test state: RFC2544_TEST_STATE_COMPLETED
       Test start time: 2005-07-29 10:36:54 UTC
       Test finish time: 2005-07-29 10:37:57 UTC
```

```
      Counters last cleared: Never

      Test-profile Configuration:
          Test-profile name: prof_b2b
          Test packet size: 64 1280
          Therotical max bandwidth : 993000 kbps

      Test Configuration:
          Test mode: Initiate-and-Terminate
          Duration in seconds: 20
          Test family: INET
          Routing Instance Name: default

      Inet family Configuration:
          Egress Interface : ge-0/1/1.0
          Source ipv4 address: 192.0.2.1
          Destination ipv4 address: 192.0.2.2
          Source udp port: 2020
          Destination udp port: 3030

  Rfc2544 Back-Back test information :
      Initial burst length: 20 seconds at 993000 kbps
      Test iteration mode : Binary
      Test iteration step percent : 50.00 %

Test packet size: 64
Iteration   Burst Length    Elapsed         Tx           Rx              Tx
       Rx
             (Packets)       time        Packets       Packets         Bytes
     Bytes
    1          404754            3        404754        404754       27523272
    27523272
    2        26983680           20      26983680      26983680     1834890240
  1834890240

Result of the iteration runs : Back-Back-Frames Test complete for packet size 64

Best iteration : 2
Measured burst (num sec)  : 20 sec,
Measured burst (num pkts)  : 26983680 packets
Result of the iteration runs : Back-Back-Frames Test complete for packet size 64

Best iteration : 2
Measured burst (num sec)  : 20 sec,
```

```
Measured burst (num pkts)  : 26983680 packets

Test packet size: 1280
Iteration   Burst Length    Elapsed         Tx           Rx              Tx
       Rx
             (Packets)       time        Packets       Packets         Bytes
     Bytes
    1           28467            3         28467         28467       36551628
    36551628
    2         1897920           20       1897920       1897920     2436929280
  2436929280

Result of the iteration runs : Back-Back-Frames Test complete for packet size 12

Best iteration : 2
Measured burst (num sec)  : 20 sec,
Measured burst (num pkts)  : 1897920 packets

RFC2544 Back-Back test results summary:
----------------------------------------

  Packet      Measure Burst
   Size      length (Packets)
     64       26983680 packets
   1280       1897920 packets
```

show services rpm rfc2544-benchmarking test-id detail (Reflection Test on MX104 Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id detail 1**

```
 Test information :
     Test id: 1, Test name: fort_uni_inet_ref, Test type: Reflect
     Test mode: Reflect
     Test packet size: 0
     Test state: RFC2544_TEST_STATE_RUNNING
     Status: Running
     Test start time: 2013-12-09 16:24:52 IST
     Test finish time: TEST_RUNNING
     Counters last cleared: Never

     Test Configuration:
         Test mode: Reflect
         Duration in seconds: 864000
```

```
          Test family: INET
          Routing Instance Name: default

      Inet family Configuration:
          Egress Interface : ge-0/3/1.0
          Destination ipv4 address: 198.51.100.2
          Destination udp port: 200

Elapsed        Reflected            Reflected
 time           Packets              Bytes
    176         8977917            9031784502
```

show services rpm rfc2544-benchmarking test-id brief (Reflection Test on MX104 Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id brief 1**

```
  Test information :
      Test id: 1, Test name: fort_uni_inet_ref, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: RFC2544_TEST_STATE_RUNNING
      Status: Running
      Test start time: 2013-12-09 16:24:52 IST
      Test finish time: TEST_RUNNING
      Counters last cleared: Never
```

show services rpm rfc2544-benchmarking test-id detail (Reflection Test on MX104 Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id detail 2**

```
  Test information :
      Test id: 2, Test name: fort_uni_inet_ref, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: RFC2544_TEST_STATE_RUNNING
      Status: Running
      Test start time: 2013-12-09 16:39:18 IST
      Test finish time: TEST_RUNNING
      Counters last cleared: Never

      Test Configuration:
          Test mode: Reflect
          Duration in seconds: 864000
```

```
          Test family: CCC
          Routing Instance Name: default

      CCC family Configuration:
          Interface : ge-0/3/2.0
          Test direction: Egress

Elapsed        Reflected            Reflected
 time           Packets              Bytes
     23          809137             825319740
```

show services rpm rfc2544-benchmarking test-id brief (Reflection Test on MX104 Routers)

`user@host>` **show services rpm rfc2544-benchmarking test-id 2 brief**

```
  Test information :
      Test id: 2, Test name: fort_uni_inet_ref, Test type: Reflect
      Test mode: Reflect
      Test packet size: 0
      Test state: RFC2544_TEST_STATE_RUNNING
      Status: Running
      Test start time: 2013-12-09 16:39:18 IST
      Test finish time: TEST_RUNNING
      Counters last cleared: Never
```

show services rpm twamp client connection

Syntax

```
show services rpm twamp client connection
<connection-name>
```

Release Information

Command introduced in Junos OS Release 15.1 for MX Series routers.

Description

Display information about the connections established between the real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) server and control-clients. By default, all established sessions are displayed, unless you specify a control-connection name when you issue the command.

Options

none—Display information about all TWAMP client connection sessions.

connection-name—(Optional) Display information about the specified control-connection or TWAMP control-client.

Required Privilege Level

view

List of Sample Output

show services rpm twamp client connection on page 1400

Output Fields

Table 143 on page 1399 lists the output fields for the **show services rpm twamp client connection** command. Output fields are listed in the approximate order in which they appear.

Table 143: show services rpm twamp client connection Output Fields

Field Name	Field Description
Connection Name	Connection name that uniquely identifies the connection between the TWAMP server and a particular client.
Client address	Client IP address.
Client port	Client port number.
Server address	Server IP address.

Table 143: show services rpm twamp client connection Output Fields *(continued)*

Field Name	Field Description
Server port	Server port number.
Session count	Session count.
Auth mode	Authentication mode.

Sample Output

show services rpm twamp client connection

`user@host>` **show services rpm twamp client connection**

```
Connection  Client       Client  Server          Server   Session  Auth
   ID       address       port   address          port     count   mode
    4     192.0.2.1       12345  192.168.219.203  890        16    none
   78   198.51.100.55     345    203.0.113.2      89022       5    none
  234   192.168.219.203   2345   192.168.22.2     3333       16    none
    5   192.168.3.1       82345  192.168.2.2      45909      16    authenticated

    1   192.168.1.1       645    192.168.4.23     2394       16    encrypted
```

show services rpm twamp client history-results

Syntax

```
show services rpm history-results
<brief | detail>
<control-connection control-connection-name>
<since time>
<test-session test-session-name>
```

Release Information

Command introduced in Junos OS Release 15.1 for MX Series routers.

Description

Display standard information about the results of the last 50 probes for each real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) instance. You can also view the historical results of the probes or test packets sent from a TWAMP client to a TWAMP server for a particular control-connection, or a test-session associated with a control-connection.

Options

none—Display the results of the last 50 probes for all RPM TWAMP instances.

brief | detail—(Optional) Display the specified level of output.

control-connection *control-connection-name*—(Optional) Display information for the specified control-connection between a TWAMP client and a TWAMP server.

since *time*—(Optional) Display information from the specified time. Specify time as ***yyyy-mm-dd.hh*:*mm*:*ss***.

test-session *test-session-name*—(Optional) Display information for the specified test session associated with a control-connection between a TWAMP client and a TWAMP server.

Required Privilege Level

view

List of Sample Output

show services rpm twamp client history-results on page 1403
show services rpm twamp client history-results detail on page 1404

Output Fields

Table 139 on page 1353 lists the output fields for the **show services rpm twamp client history-results** command. Output fields are listed in the approximate order in which they appear.

Table 144: show services rpm twamp client history-results Output Fields

Field Name	Field Description	Level of Output
Owner	Probe owner or the TWAMP client.	All levels
Test	Name of a test for a TWAMP probe instance.	All levels
Probe received	Timestamp when the probe result was determined.	All levels
Round trip time	Average ping round-trip time (RTT), in microseconds.	All levels
Probe results	Result of a particular probe performed by a remote host. The following information is contained in the results: • **Response received**—Timestamp when the probe result was determined. • **Rtt**—Average ping round-trip time (RTT), in microseconds.	**detail**
Results over current test	Displays the results for the current test by probe at the time each probe was completed, as well as the status of the current test at the time the probe was completed.	**detail**
Probes sent	Number of probes sent with the current test.	**detail**
Probes received	Number of probe responses received within the current test.	**detail**
Loss percentage	Percentage of lost probes for the current test.	**detail**
Measurement	Increment of measurement. Possible values are round-trip time delay and, for the probe type icmp-ping-timestamp, the egress and ingress delay: • **Minimum**—Minimum RTT, ingress delay, or egress delay measured over the course of the current test. • **Maximum**—Maximum RTT, ingress delay, or egress delay measured over the course of the current test. • **Average**—Average RTT, ingress delay, or egress delay measured over the course of the current test. • **Peak to peak**—Difference between two peak values of RTT, ingress delay, or egress delay measured over the course of the current test. • **Jitter**—Difference, in microseconds, between the maximum and minimum RTT measured over the course of the current test. • **Sum**—Total round-trip time, in microseconds, measured over the course of the current test.	**detail**

Sample Output

show services rpm twamp client history-results

user@host> **show services rpm twamp client history-results**

```
Owner, Test                 Probe received              Round trip time
c2, t1                      Fri Feb 13 00:31:54 2015  Request timed out
c2, t1                      Fri Feb 13 00:31:55 2015  Request timed out
c2, t1                      Fri Feb 13 00:31:56 2015  Request timed out
c2, t1                      Fri Feb 13 00:31:57 2015  Request timed out
c2, t1                      Fri Feb 13 00:31:58 2015  Request timed out
c2, t1                      Fri Feb 13 00:31:59 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:00 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:01 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:02 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:03 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:04 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:05 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:06 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:07 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:08 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:09 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:10 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:11 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:12 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:13 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:14 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:15 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:16 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:17 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:18 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:19 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:20 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:21 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:22 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:23 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:24 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:25 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:26 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:27 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:28 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:29 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:30 2015  Request timed out
c2, t1                      Fri Feb 13 00:32:31 2015  Request timed out
```

```
    c2, t1                      Fri Feb 13 00:32:32 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:33 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:34 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:35 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:36 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:37 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:38 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:39 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:40 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:41 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:42 2015  Request timed out
    c2, t1                      Fri Feb 13 00:32:43 2015  Request timed out

p1, t1                      Wed Aug 12 01:02:42 2009            1180 usec
```

show services rpm twamp client history-results detail

user@host> **show services rpm twamp-client history-results detail**

```
    Owner: p, Test: t
    Probe results:
      Response received, Tue Jan  7 05:11:49 2014,
      Rtt: 184 usec, Round trip jitter: -96 usec, Round trip interarrival jitter:
 57 usec
    Results over current test:
      Probes sent: 4, Probes received: 4, Loss percentage: 0
      Measurement: Round trip time
        Samples: 4, Minimum: 174 usec, Maximum: 196 usec, Average: 183 usec, Peak
 to peak: 22 usec, Stddev: 8 usec, Sum: 732 usec
      Measurement: Positive round trip jitter
        Samples: 1, Minimum: 110 usec, Maximum: 110 usec, Average: 110 usec, Peak
 to peak: 0 usec, Stddev: 0 usec, Sum: 110 usec
      Measurement: Negative round trip jitter
        Samples: 2, Minimum: 96 usec, Maximum: 811 usec, Average: 454 usec, Peak
to peak: 715 usec, Stddev: 358 usec, Sum: 907 usec

    Owner: p, Test: t
    Probe results:
      Response received, Tue Jan  7 05:11:50 2014,      Rtt: 174 usec, Round trip
 jitter: -8 usec, Round trip interarrival jitter: 54 usec
    Results over current test:
      Probes sent: 5, Probes received: 5, Loss percentage: 0
      Measurement: Round trip time
```

```
       Samples: 5, Minimum: 174 usec, Maximum: 196 usec, Average: 181 usec, Peak
 to peak: 22 usec, Stddev: 8 usec, Sum: 906 usec
     Measurement: Positive round trip jitter
       Samples: 1, Minimum: 110 usec, Maximum: 110 usec, Average: 110 usec, Peak
 to peak: 0 usec, Stddev: 0 usec, Sum: 110 usec
     Measurement: Negative round trip jitter
       Samples: 3, Minimum: 8 usec, Maximum: 811 usec, Average: 305 usec, Peak
to peak: 803 usec, Stddev: 360 usec, Sum: 915 usec
```

show services rpm twamp client probe-results

Syntax

```
show services rpm twamp client probe-results
<control-connection control-connection-name>
<test-session test-session-name>
```

Release Information

Command introduced in Junos OS Release 15.1 for MX Series routers.

Description

Display the results of the most recent real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) probes sent from the TWAMP client to the TWAMP server. You can also view the results of the probes or test packets sent from a TWAMP client to a TWAMP server for a particular control-connection, or a test-session associated with a control-connection.

Options

none—Display all results of the most recent TWAMP probes.

control-connection *control-connection-name*—(Optional) Display information for the specified control-connection between a TWAMP client and a TWAMP server.

test-session *test-session-name*—(Optional) Display information for the specified test session associated with a control-connection between a TWAMP client and a TWAMP server.

Required Privilege Level

view

List of Sample Output

show services rpm twamp client probe-results on page 1410
show services rpm twamp client probe-results control-connection on page 1411

Output Fields

Table 145 on page 1406 lists the output fields for the **show services twamp client probe-results** command. Output fields are listed in the approximate order in which they appear.

Table 145: show services twamp client probe-results Output Fields

Field Name	Field Description
Owner	Name of the session-sender or the control-client, which is the TWAMP client. When you configure the ***control-client-name*** option at the **[edit services twamp client control-connection]** hierarchy level, this field displays the configured owner name or the client name.

Table 145: show services twamp client probe-results Output Fields *(continued)*

Field Name	Field Description
Test	Name of a test representing a collection of probes. When you configure the **test-session *test-name*** statement at the **[edit services owner]** hierarchy level, the field displays the configured test name.
server-address	Destination address used for the probes.
server-port	Destination port used for the probes.
Client address	Source or TWAMP client address used for the probes.
Client port	Source or TWAMP client port used for the probes.
Reflector address	Session-reflector or TWAMP server address used for the probes.
Reflector port	Session-reflector or TWAMP server port used for the probes.
Probe type	Protocol configured on the receiving probe server: **http-get, http-metadata-get, icmp-ping, icmp-ping-timestamp, tcp-ping, udp-ping,** or **udp-ping-timestamp**.
Test size	Number of probes within a test.
Destination Interface Name	Name of the interface configured on the TWAMP server or the session-reflector on which the TWAMP probe packets sent from the TWAMP client are received.
Probe results	Raw measurement of a particular probe sample done by a remote host. This data is provided separately from the calculated results. The following information is contained in the raw measurement: • **Response received**—Timestamp when the probe result was determined. • **Client and server hardware timestamps**—If timestamps are configured, an entry appears at this point. • **Rtt**—Average ping round-trip time (RTT), in microseconds. • **Egress jitter**—Egress jitter, in microseconds. • **Ingress jitter**—Ingress jitter, in microseconds. • **Round trip jitter**—Round-trip jitter, in microseconds. • **Egress interarrival jitter**—Egress interarrival jitter, in microseconds. • **Ingress interarrival jitter**—Ingress interarrival jitter, in microseconds. • **Round trip interarrival jitter**—Round-trip interarrival jitter, in microseconds.

Table 145: show services twamp client probe-results Output Fields *(continued)*

Field Name	Field Description
Results over current test	Probes are grouped into tests, and the statistics are calculated for each test. If a test contains 10 probes, the average, minimum, and maximum results are calculated from the results of those 10 probes. If the command is issued while the test is in progress, the statistics use information from the completed probes. • **Probes sent**—Number of probes sent within the current test. • **Probes received**—Number of probe responses received within the current test. • **Loss percentage**—Percentage of lost probes for the current test. • **Measurement**—Measurement type. Possible values are round-trip time, positive round-trip jitter, negative round-trip jitter, egress time, positive egress jitter, negative egress jitter, ingress time, positive ingress jitter, negative ingress jitter, and, for the probe type **icmp-ping-timestamp**, the egress delay and ingress delay. For each measurement type, the following individual calculated results are provided: ◦ **Samples**—Number of probes. ◦ **Minimum**—Minimum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Maximum**—Maximum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Average**—Average RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Peak to peak**—Peak-to-peak difference, in microseconds. ◦ **Stddev**—Standard deviation, in microseconds. ◦ **Sum**—Statistical sum.

Table 145: show services twamp client probe-results Output Fields *(continued)*

Field Name	Field Description
Results over last test	Results for the most recently completed test. If the command is issued while the first test is in progress, this information is not displayed • **Probes sent**—Number of probes sent for the most recently completed test. • **Probes received**—Number of probe responses received for the most recently completed test. • **Loss percentage**—Percentage of lost probes for the most recently completed test. • **Test completed**—Time the most recent test was completed. • **Measurement**—Measurement type. Possible values are round-trip time, positive round-trip jitter, negative round-trip jitter, egress time, positive egress jitter, negative egress jitter, ingress time, positive ingress jitter, negative ingress jitter, and, for the probe type **icmp-ping-timestamp**, the egress delay and ingress delay. For each measurement type, the following individual calculated results are provided: ◦ **Samples**—Number of probes. ◦ **Minimum**—Minimum RTT, ingress delay, or egress delay measured for the most recently completed test. ◦ **Maximum**—Maximum RTT, ingress delay, or egress delay measured for the most recently completed test. ◦ **Average**—Average RTT, ingress delay, or egress delay measured for the most recently completed test. ◦ **Peak to peak**—Peak-to-peak difference, in microseconds. ◦ **Stddev**—Standard deviation, in microseconds. ◦ **Sum**—Statistical sum.

Table 145: show services twamp client probe-results Output Fields *(continued)*

Field Name	Field Description
Results over all tests	Displays statistics made for all the probes, independently of the grouping into tests, as well as statistics for the current test. • **Probes sent**—Number of probes sent in all tests. • **Probes received**—Number of probe responses received in all tests. • **Loss percentage**—Percentage of lost probes in all tests. • **Measurement**—Measurement type. Possible values are round-trip time, positive round-trip jitter, negative round-trip jitter, egress time, positive egress jitter, negative egress jitter, ingress time, positive ingress jitter, negative ingress jitter, and, for the probe types **icmp-ping-timestamp** and **udp-ping-timestamp**, the egress delay and ingress delay. For each measurement type, the following individual calculated results are provided: ◦ **Samples**—Number of probes. ◦ **Minimum**—Minimum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Maximum**—Maximum RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Average**—Average RTT, ingress delay, or egress delay measured over the course of the current test. ◦ **Peak to peak**—Peak-to-peak difference, in microseconds. ◦ **Stddev**—Standard deviation, in microseconds. ◦ **Sum**—Statistical sum.

Sample Output

show services rpm twamp client probe-results

`user@host>` **show services rpm twamp client probe-results**

```
Owner: c2, Test: t1
server-address: 192.0.2.14, server-port: 862, Client address: 192.0.2.13,
Client port: 57170
Reflector address: 192.0.2.14, Reflector port: 10011,
Sender address: 192.0.2.13, sender-port: 10011
Destination interface name: si-1/1/0.10
Test size: 500 probes
Probe results:
  Request timed out, Fri Feb 13 00:18:29 2015
```

```
Results over current test:
  Probes sent: 349, Probes received: 0, Loss percentage: 100.00000
Results over last test:
  Probes sent: 500, Probes received: 0, Loss percentage: 100.00000
Results over all tests:
  Probes sent: 4349, Probes received: 0, Loss percentage: 100.00000
```

show services rpm twamp client probe-results control-connection

user@host> **show services rpm twamp client probe-results control-connection c2**

```
Owner: c2, Test: t1
server-address: 192.0.2.14, server-port: 862, Client address: 192.0.2.13,
Client port: 57170
Reflector address: 192.0.2.14, Reflector port: 10010,
Sender address: 192.0.2.13, sender-port: 10010
Destination interface name: si-1/1/0.10
Test size: 500 probes
Probe results:
  Request timed out, Fri Feb 13 00:07:14 2015
Results over current test:
  Probes sent: 188, Probes received: 0, Loss percentage: 100.00000
Results over last test:
  Probes sent: 500, Probes received: 0, Loss percentage: 100.00000
Results over all tests:
  Probes sent: 3688, Probes received: 0, Loss percentage: 100.00000
```

show services rpm twamp client session

Syntax

```
show services rpm twamp client session
<control-connection control-connection-name>
<test-session test-session-name>
```

Release Information

Command introduced in Junos OS Release 15.1 for MX Series routers.

Description

Display information about the sessions established between the real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) server and control clients for control packets and data packets. By default, all established control-connection and data-connection or test sessions are displayed, unless you specify a control-connection name or a test-session name when you issue the command.

Options

none—Display information about all established connections and sessions.

control-connection *control-connection-name*—(Optional) Display information about the specified control-connection, which is established for control-packets exchanged between a TWAMP client and a TWAMP server.

test-session *test-session-name*—(Optional) Display information about the specified test session, which is established for data packets transmitted between a TWAMP client and a TWAMP server, associated with a control-connection..

Required Privilege Level

view

List of Sample Output

show services rpm twamp client session on page 1413
show services rpm twamp client session control-connection on page 1413

Output Fields

Table 146 on page 1412 lists the output fields for the **show services rpm twamp client session** command. Output fields are listed in the approximate order in which they appear.

Table 146: show services rpm twamp client session Output Fields

Field Name	Field Description
Connection Name	Name of the control connection that uniquely identifies the connection between the TWAMP server and the TWAMP client.

Table 146: show services rpm twamp client session Output Fields *(continued)*

Field Name	Field Description
Session Name	Name of the test session that uniquely identifies the data-session between the TWAMP server and the TWAMP client.
Sender address	Sender IP address.
Sender port	Sender port number.
Reflector address	Reflector IP address.
Reflector port	Reflector port number.

Sample Output

show services rpm twamp client session

`user@host>` **show services rpm twamp client session**

```
Connection  Session   Sender       Sender    Reflector    Reflector
      Name     Name   address       port      address       port
      cs1       ts1   198.51.100.1  41998   198.51.100.2    5008
      cs2       ts1   198.51.100.1  53710   198.51.100.2    5009
```

show services rpm twamp client session control-connection

`user@host>` **show services rpm twamp client session control-connection c2**

```
Connection  Session   Sender        Sender    Reflector  Reflector
      Name     Name   address         port      address     port
        c2       t1   192.0.2.13     10008   192.0.2.14    10008
```

show services rpm twamp server connection

Syntax

```
show services rpm twamp server connection
<connection-id>
```

Release Information

Command introduced in Junos OS Release 9.3.

Description

Display information about the connections established between the real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) server and control-clients. By default, all established sessions are displayed, unless you specify a session ID when you issue the command.

Options

none—Display connection information about all established sessions.

connection-id—(Optional) Identifier of the connection that you want to display information about.

Required Privilege Level

view

List of Sample Output

show services rpm twamp server connection on page 1415

Output Fields

Table 143 on page 1399 lists the output fields for the **show services rpm twamp server connection** command. Output fields are listed in the approximate order in which they appear.

Table 147: show services rpm twamp server connection Output Fields

Field Name	Field Description
Connection ID	Connection ID that uniquely identifies the connection between the TWAMP server and a particular client.
Client address	Client IP address.
Client port	Client port number.
Server address	Server IP address.
Server port	Server port number.

Table 147: show services rpm twamp server connection Output Fields *(continued)*

Field Name	Field Description
Session count	Session count.
Auth mode	Authentication mode.

Sample Output

show services rpm twamp server connection

`user@host>` **show services rpm twamp server connection**

```
Connection  Client       Client  Server           Server  Session Auth
ID          address      port    address          port    count   mode
 4          192.0.2.1    12345  192.168.219.203  890     16      none
78         198.51.100.55  345    203.0.113.2      89022   5       none
234      192.168.219.203 2345   192.168.22.2     3333    16      none
 5         192.168.3.1    82345  192.168.2.2      45909   16   authenticated
 1         192.168.1.1    645    192.168.4.23     2394    16   encrypted
```

show services rpm twamp server session

Syntax

```
show services rpm twamp server session
<session-id>
```

Release Information

Command introduced in Junos OS Release 9.3.

Description

Display information about the sessions established between the real-time performance monitoring (RPM) Two-Way Active Measurement Protocol (TWAMP) server and control clients. By default, all established sessions are displayed, unless you specify a session ID when you issue the command.

Options

none—Display information about all established sessions.

session-id—(Optional) Identifier of the session that you want to display information about.

Required Privilege Level

view

List of Sample Output

show services rpm twamp server session on page 1417

Output Fields

Table 148 on page 1416 lists the output fields for the **show services rpm twamp server session** command. Output fields are listed in the approximate order in which they appear.

Table 148: show services rpm twamp server session Output Fields

Field Name	Field Description
Session ID	Session ID that uniquely identifies the session between the TWAMP server and a particular client.
Connection ID	Connection ID that uniquely identifies the connection between the TWAMP server and a particular client.
Sender address	Sender IP address.
Sender port	Sender port number.
Reflector address	Reflector IP address.

Table 148: show services rpm twamp server session Output Fields *(continued)*

Field Name	Field Description
Reflector port	Reflector port number.

Sample Output

show services rpm twamp server session

`user@host>` **show services rpm twamp server session**

```
    Session  Connection  Sender          Sender  Reflector      Reflector
    ID       ID          address         port    address        port
     4       44         192.0.2.1        12345  192.168.219.203    890
    78       44         198.51.100.55    345    203.0.113.2       89022
   234       423        192.168.219.203  2345   192.168.22.2       3333
     5       423        192.168.3.1      82345  192.168.2.2       45909
     1       423        192.168.1.1      645    192.168.4.23       2394
```

show services service-sets statistics jflow-log

Syntax

```
show services service-sets statistics jflow-log
<interface interface-name>
<service-set service-set-name>
<brief | detail>
```

Release Information

Command introduced in Junos OS Release 15.1.

Description

Display statistical information on the logs or records generated in flow monitoring format with optional filtering by interface and service set name..

Options

none—Display the statistical details on flow monitoring logs for NAT events for all services interfaces and all service sets.

brief—(Default) (Optional) Display abbreviated flow monitoring log statistics.

detail—(Optional) Display detailed flow monitoring log statistics.

interface *interface-name*—(Optional) Display the flow monitoring log statistics for the specified adaptive service interface. On M Series and T Series routers, ***interface-name*** can be **ms-*fpc/pic/port***. It is supported only on MS-MICs and MS-MPCs.

service-set *service-set name*—(Optional) Display the flow monitoring log statistics for the specified named service-set.

Required Privilege Level

view

RELATED DOCUMENTATION

clear services service-sets statistics syslog

List of Sample Output

show services service-sets statistics jflow-log brief on page 1422
show services service-sets statistics jflow-log detail on page 1423
show services service-sets statistics jflow-log service-set on page 1425
show services service-sets statistics jflow-log service-set detail on page 1425

Output Fields

Table 149 on page 1419 lists the output fields for the **show services service-sets statistics jflow-log** command. Output fields are listed in the approximate order in which they appear.

Table 149: show services service-sets statistics jflow-log Output Fields

Field Name	Field Description	Level of Output
Interface	Name of a services interface.	all
Rate limit	Maximum number of NAT error events for which records in flow monitoring format must be recorded.	all
Template records	Details of the template records in flow monitoring log messages.	all
Sent	Number of template records sent to a collector	all
Messages dropped	Number of template records dropped while transmission to a collector.	all
Data records	Details of the data records in flow monitoring log messages.	all
Sent	Number of data records sent to a collector.	all
Dropped	Number of data records dropped while transmission to a collector	all
Service set	Name of a service set.	all
Unresolvable collectors	Number of collectors that cannot be traced and be reached to export records for NAT events.	all

Table 149: show services service-sets statistics jflow-log Output Fields *(continued)*

Field Name	Field Description	Level of Output
class name		**detail**

Table 149: show services service-sets statistics jflow-log Output Fields *(continued)*

Field Name	Field Description	Level of Output
	Logs created for events for each of the following classes: • **NAT44 Session logs**—Details of logs created for NAT44 sessions • **NAT64 Session logs**—Details of logs created for NAT64 sessions • **NAT44 BIB logs**—Details of logs created for NAT44 binding information bases, which is a table of bindings kept by a NAT44. Each NAT44 has a BIB for each translated protocol. • **NAT64 BIB logs**—Details of logs created for NAT44 binding information bases, which is a table of bindings kept by a NAT64. Each NAT64 has a BIB for each translated protocol. • **NAT Address Exhausted logs**—Details of logs created for exhaustion of NAT addresses • **NAT Port Exhausted logs**—Details of logs created for exhaustion of NAT pool • **NAT44 Quota Exceeded logs**—Details of logs created when allocated quota is exceeded for NAT44 events • **NAT64 Quota Exceeded logs**—Details of logs created when allocated quota is exceeded for NAT64 events • **NAT44 Address Bind logs**—Details of logs generated for address bindings for NAT44 events • **NAT64 Address Bind logs**—Details of logs generated for address bindings for NAT64 events • **NAT44 PBA logs**—Details of logs generated for NAT44 port block allocation events • **NAT64 PBA logs**—Details of logs generated for NAT64 port block allocation events The following information is displayed for flow monitoring log messages for each class of event that is logged: • **Template records**—Details of the template records in flow monitoring log messages • **Sent**—Number of template records sent to a collector • **Dropped**—Number of template records dropped while transmission to a collector • **Data records**—Details of the data records in flow monitoring log messages • **Sent**—Number of data records sent to a collector	

Table 149: show services service-sets statistics jflow-log Output Fields *(continued)*

Field Name	Field Description	Level of Output
	• **Dropped**—Number of data records dropped while transmission to a collector. Counts are provided for the drop reasons • **socket send error**—Number of times a socket was not opened for data transmission • **no memory**—Number of messages dropped because of insufficient memory • **invalid data**—Number of messages dropped because of invalid data in the records • **above rate limit**—The maximum number of flow monitoring log messages per second was exceeded.	

Sample Output

show services service-sets statistics jflow-log brief

```
user@host> show services service-sets statistics jflow-log brief
```

```
Interface: ms-5/0/0
  Rate limit: 1000
  Template records:
    Sent: 36
    Dropped: 0
  Data records:
    Sent: 2
    Dropped: 0

    Service-set: sset_44
      Unresolvable collectors: 0
      Template records:
        Sent: 36
        Dropped: 0
      Data records:
        Sent: 2
        Dropped: 0
```

show services service-sets statistics jflow-log detail

user@host> **show services service-sets statistics jflow-log detail**

```
Interface: ms-5/0/0
  Rate limit: 1000
  Template records:
    Sent: 48
    Dropped: 0
  Data records:
    Sent: 4
    Dropped: 0

    Service-set: sset_44
      Unresolvable collectors: 0
      Template records:
        Sent: 48
        Dropped: 0
      Data records:
        Sent: 4
        Dropped: 0
      NAT44 Session logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 4
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT64 Session logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 0
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT44 BIB logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 0
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT64 BIB logs:
        Template records:
          Sent: 4
```

```
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT Address Exhausted logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT Port Exhausted logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 Quota Exceeded logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 Quota Exceeded logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 Address Bind logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 Address Bind logs:
  Template records:
    Sent: 4
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
```

```
          Sent: 0
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT44 PBA logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 0
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
      NAT64 PBA logs:
        Template records:
          Sent: 4
          Dropped: 0 (socket send error: 0, no memory: 0)
        Data records:
          Sent: 0
          Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
```

show services service-sets statistics jflow-log service-set

user@host> **show services service-sets statistics jflow-log service-set sset_44**

```
Interface: ms-5/0/0

    Service-set: sset_44
      Unresolvable collectors: 0
      Template records:
        Sent: 72
        Dropped: 0
      Data records:
        Sent: 4
        Dropped: 0
```

show services service-sets statistics jflow-log service-set detail

user@host> **show services service-sets statistics jflow-log service-set sset_44 detail**

```
Interface: ms-5/0/0

    Service-set: sset_44
      Unresolvable collectors: 0
      Template records:
        Sent: 84
```

```
  Dropped: 0
Data records:
  Sent: 4
  Dropped: 0
NAT44 Session logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 4
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 Session logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 BIB logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 BIB logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT Address Exhausted logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT Port Exhausted logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
```

```
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 Quota Exceeded logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 Quota Exceeded logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 Address Bind logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 Address Bind logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT44 PBA logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
NAT64 PBA logs:
  Template records:
    Sent: 7
    Dropped: 0 (socket send error: 0, no memory: 0)
  Data records:
    Sent: 0
    Dropped: 0 (invalid data: 0, no memory: 0, above rate limit: 0)
```

show services video-monitoring mdi errors fpc-slot

Syntax

```
show services video-monitoring mdi errors fpc-slot fpc-slot
```

Release Information

Command introduced in Junos OS Release 14.1.

Description

Display video monitoring error statistics.

Options

fpc-slot—Number of the fpc slot for which statistics are displayed.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

List of Sample Output

show services video-monitoring mdi errors fpc-slot on page 1429

Output Fields

Table 110 on page 847 lists the output fields for the **show services video-monitoring mdi errors fpc-slot** command. Output fields are listed in the approximate order in which they appear.

Table 150: show services video-monitoring mdi errors fpc-slot Output Fields

Field Name	Field Description
FPC slot	Slot number of the monitored FPC.
Flow Insert Error	Number of errors during new flow insert operations.

Table 150: show services video-monitoring mdi errors fpc-slot Output Fields *(continued)*

Field Name	Field Description
Flow Policer Drops	Number of packets dropped by flow policer process. **NOTE:** New flows usually arrive within a very short time interval (1.5 microseconds). These errors do not represent the loss of entire flows, because subsequent packets in the flow can establish the flow. All packets are monitored after a flow has been established. Packet forwarding occurs independently of the video monitoring, and packets are not dropped due to video monitoring errors.
Unsupported Media Packets Count	Number of packets dropped because they are not media packets or they are unsupported media packets.
PID Limit Exceeded	Number of packets unmonitored because the process identifier (PID) limit exceeded has been exceeded. **NOTE:** The current PID limit is 6.

Sample Output

show services video-monitoring mdi errors fpc-slot

`user@host>` **show services video-monitoring mdi errors fpc-slot 2**

```
MDI Errors Information
   FPC Slot: 2
   Flow Insert Error: 0, Flow Policer Drops: 0
   Unsupported Media Packets Count: 0, PID Limit Exceeded: 202995
```

show services video-monitoring mdi flows fpc-slot

Syntax

```
show services video-monitoring mdi flows fpc-slot fpc-slot
<brief>
<count>
<destination-address>
<destination-port>
<detail>
<df-mlr-split-view>
<flow-over-ipv4 | flow-over-ipv4-over-mpls | flow-over ipv6 | flow-over-ipv6-mpls>
<input>
<interface-name>
<output>
<rtp>
<source-address>
<source-port>
<template-name>
<udp>
```

Release Information

Command introduced in Junos OS Release 14.1.

Description

Display inline video monitoring flow statistics.

Options

fpc-slot—Number of the slot for which flows are reported.

brief—(Optional) Display brief output (default).

count—(Optional) Display the number of flows.

destination-address—(Optional) Filter output by destination address.

destination-port—(Optional) Filter output by destination port.

detail—(Optional) Display output in detailed format including media delivery index records.

df-mlr-split-view—(Optional) Display detailed/brief flow output with DF and MLR split.

flow-over-ipv4 | flow-over-ipv4-over-mpls | flow-over ipv6 | flow-over-ipv6-mpls—(Optional) Display only IPv4 flows, only IPv4-over-MPLS flows, only IPv6 flows, or only IPv6-over-MPLS flows.

input—(Optional) Filter output by flow direction input.

interface-name—(Optional) Filter output by logical interface name.

output—(Optional) Filter output by flow direction output.

rtp—(Optional) Filter output by flow type rtp.

source-address—(Optional) Filter output by source IP address.

source-port—(Optional) Filter output by source port.

template-name—(Optional) Filter output by media delivery index template name.

udp—(Optional) Filter output by flow type MPEG-TS.

df-mlr-split-view—(Optional) Display detailed/brief flow output with DF and MLR split.

Required Privilege Level
view

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

List of Sample Output
show services video-monitoring mdi flows fpc-slot brief on page 1433
show services video-monitoring mdi flows fpc-slot 0 detail on page 1433
show services video-monitoring mdi flows fpc-slot 2 detail on page 1434

Output Fields
Table 111 on page 848 lists the output fields for the **show services video-monitoring mdi flows fpc-slot** command. Output fields are listed in the approximate order in which they appear.

Table 151: show services video-monitoring mdi flows fpc-slot Output Fields

Field Name	Field Description
SIP	Source IP address.
DIP	Destination IP address.
SP	Source port.
DP	Destination port.
Di	Direction of flow (I=Input, O=Output).

Table 151: show services video-monitoring mdi flows fpc-slot Output Fields *(continued)*

Field Name	Field Description
Ty	Type of flow.
Last DF:MLR	Delay factor and media loss rate value of last media delivery index record. NOTE: If you choose **df-mlr-split-view** option, then the DF and MLR values will be displayed in split format as: **Last DF:<*xxx*>, Last MLR:<*xxx*>, Avg DF:<*xxx*>, Avg MLR:<*xxx*>**
Avg DF:MLR	Average value of delay factor and media loss rate.
Last MRV	Media rate variation value of last media delivery index record.
Avg MRV	Average value of media rate variation.
IFL	Interface name on which flow is received.
Template Name	Name of template associated with flow.
Flow Identifier	Identifier for the flow.
Source Address	Source IP address.
Destination Address	Destination IP address.
Interface Name	Interface name on which flow is received.
Flow Direction	Direction of flow (I=Input, O=Output).
Flow Type	Flow type that is monitored. The value is one of the following: • RTP • RTP over IPv6 • RTP over IPv4-over-MPLS • RTP over IPv6-over-MPLS • UDP • UDP over IPv6 • UDP over IPv4-over-MPLS • UDP over IPv6-over-MPLS

Table 151: show services video-monitoring mdi flows fpc-slot Output Fields *(continued)*

Field Name	Field Description
Rec No	Record number.
PID	Process identifier.
MLR	Media loss rate.

Sample Output

show services video-monitoring mdi flows fpc-slot brief

`user@host>` **show services video-monitoring mdi flows fpc-slot 2 brief**

```
---------------------------------------------------------------------------------------------------------------------------
Sno |SIP      |SP  |DIP         |DP  |Di|Ty |Last DF:MLR |Avg DF:MLR |Last MRV
|Avg MRV |IFL         |Template Name |Flow Identifier

---------------------------------------------------------------------------------------------------------------------------
1   192.0.2.2 1024 198.51.100.2 2048 I  UDP  70.90:1      92.15:8205  -7.09
-9.36   xe-2/2/1.0        t1            16777216
```

Sample Output

show services video-monitoring mdi flows fpc-slot 0 detail

`user@host>` **show services video-monitoring mdi flows fpc-slot 0 detail**

```
Source Address: 192.0.2.22, Source Port: 1024
    Destination Address: 203.0.113.1, Destination Port: 2060
    Last DF:MLR: 3.56:0, Avg DF:MLR: 3.60:0
    Last MRV: 0.00, Avg MRV: 0.00
    Interface Name: ge-0/3/4.0, Template Name: t1
    Flow Direction: Input, Flow Type: RTP
    MDI Records Count: 10
    Flow Identifier: 16777216
--------------------------------------+
```

```
Rec No|        DF|       MLR|       MRV|
--------------------------------------+
     1       3.61          0       0.00
     2       3.64          0       0.00
     3       3.58          0       0.00
     4       3.62          0       0.00
     5       3.57          0       0.00
     6       3.60          0       0.00
     7       3.63          0       0.00
     8       3.58          0       0.00
     9       3.61          0       0.00
    10       3.56          0       0.00

Source Address: 192.0.2.22, Source Port: 1024
    Destination Address: 203.0.113.1, Destination Port: 2060
    Last DF:MLR: 3.57:0, Avg DF:MLR: 3.60:0
    Last MRV: 0.00, Avg MRV: -0.04
    Interface Name: ge-0/2/2.0, Template Name: t1
    Flow Direction: Output, Flow Type: RTP over IPv4-over-MPLS
    MPLS Labels: (299776,16,0)
    MDI Records Count: 10
    Flow Identifier: 16777217
--------------------------------------+
Rec No|        DF|       MLR|       MRV|
--------------------------------------+
     1       3.59          0       0.00
     2       3.62          0       0.00
     3       3.57          0       0.00
     4       3.60          0       0.00
     5       3.64          0       0.00
     6       3.58          0       0.00
     7       3.62          0       0.00
     8       3.57          0      -0.35
     9       3.62          0       0.00
    10       3.57          0       0.00
```

Sample Output

show services video-monitoring mdi flows fpc-slot 2 detail

`user@host>` **show services video-monitoring flows fpc-slot 2 detail count 19**

Format for RTP flows:

```
 Source Address: 192.0.2.2, Source Port: 1024
    Destination Address: 198.51.100.2, Destination Port: 2048
    Last DF:MLR: 3.58:0, Avg DF:MLR: 3.60:0
    Last MRV: 0.00, Avg MRV: 0.00
    Interface Name: xe-2/2/1.0, Template Name: t1
    Flow Direction: Input, Flow Type: RTP, MDI Records Count: 10
---------------------------------------+
Rec No|        DF|       MLR|       MRV|
---------------------------------------+
     1       3.58          0       0.00
     2       3.62          0       0.00
     3       3.59          0       0.00
     4       3.63          0       0.00
     5       3.60          0       0.00
     6       3.64          0       0.00
     7       3.61          0       0.00
     8       3.57          0       0.00
     9       3.62          0       0.00
    10       3.58          0       0.00
```

Format for MPEG2-TS over UDP flows:

```
    Source Address: 192.0.2.2, Source Port: 1024
    Destination Address: 198.51.100.2, Destination Port: 2048
    Last DF:MLR: 3.63:0, Avg DF:MLR: 3.61:4097
    Last MRV: 0.00, Avg MRV: 0.00
    Interface Name: xe-2/2/1.0, Template Name: t1
    Flow Direction: Input, Flow Type: UDP, MDI Records Count: 10
----------------------------------+-----------------------------------------------------------------------------------------+
Rec No|        DF|       MLR|       MRV|    PID-0      |    PID-1      |    PID-2
      |    PID-3      |    PID-4      |    PID-5      |
----------------------------------+-----------------------------------------------------------------------------------------+

                                       |    Val|    MLR|    Val|    MLR|    Val|
   MLR|    Val|    MLR|    Val|    MLR|    Val|    MLR|
-------------------------------------------------------------------------------------------------------------------------+

     1       3.63          0       0.00  0x1f40       0  0x1f41       0    0x12
     0  0x1f54       0    0x11       0  0x1020       0
     2       3.59          0       0.00  0x1f40       0  0x1f41       0    0x12
     0  0x1f54       0    0x11       0  0x1020       0
     3       3.64          0       0.00  0x1f40       0  0x1f41       0    0x12
     0  0x1f54       0    0x11       0  0x1020       0
```

```
 4       3.60          0       0.00  0x1f40       0  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
 5       3.64          0       0.00  0x1f40       0  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
 6       3.61          0       0.00  0x1f40       0  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
 7       3.57          0       0.00  0x1f40       0  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
 8       3.62          0       0.00  0x1f40       0  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
 9       3.58      40977       0.00  0x1f40   40977  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
10       3.63          0       0.00  0x1f40       0  0x1f41       0    0x12
 0  0x1f54       0    0x11       0  0x1020       0
```

show services video-monitoring mdi stats fpc-slot

Syntax

```
show services video-monitoring mdi stats fpc-slot fpc-slot
```

Release Information

Command introduced in Junos OS Release 14.1.

Description

Display inline video monitoring statistics.

Options

fpc-slot—Number of the fpc slot for which statistics are displayed.

Required Privilege Level

view

RELATED DOCUMENTATION

Understanding Inline Video Monitoring on MX Series Routers | 827

List of Sample Output

show services video-monitoring mdi stats fpc-slot on page 1438

Output Fields

Table 109 on page 846 lists the output fields for the **show services video-monitoring mdi stats fpc-slot** command. Output fields are listed in the approximate order in which they appear.

Table 152: show services video-monitoring mdi stats fpc-slot Output Fields

Field Name	Field Description
FPC Slot	Slot number of the monitored FPC
Active Flows	Number of active flows currently monitored. active flows = inserted flows - deleted flows.
Total Inserted Flows	Number of flows initiated under video monitoring.
Total Deleted Flows	Number of flows deleted due to inactivity timeout.

Table 152: show services video-monitoring mdi stats fpc-slot Output Fields *(continued)*

Field Name	Field Description
Total Packets Count	Number of total packets monitored.
Total Bytes Count	Number of total bytes monitored.
DF Alarm Count	Number of delay factor alarms at each of the following levels: • **Info level** • **Warning level** • **Critical level**
MLR Alarm Count	Number of media loss rate (MLR) alarms at each of the following levels: • **Info level** • **Warning level** • **Critical level**
MRV alarm count	Number of media rate variation (MRV) alarms at each of the following levels: • **Info level** • **Warning level** • **Critical level**

Sample Output

show services video-monitoring mdi stats fpc-slot

`user@host>` **show services video-monitoring mdi stats fpc-slot 2**

```
MDI Stats Information
   FPC Slot: 2
   Active Flows: 1, Total Inserted Flows: 1, Total Deleted Flows: 0
   Total Packets Count: 746284, Total Bytes Count: 1013453672
   DF alarm count: 0, Info level: 0, Warning level: 0, Critical level: 0
   MLR alarm count: 0, Info level: 0, Warning level: 0, Critical level: 0
   MRV alarm count: 0, Info level: 0, Warning level: 0, Critical level: 0
```

test services rpm rfc2544-benchmarking test

List of Syntax

Syntax (ACX Series) on page 1439
Syntax (MX104 Router) on page 1439

Syntax (ACX Series)

```
test services rpm rfc2544-benchmarking test
<clear-counters>
<routing-instance routing-instance-name>
<test-name>
<test-id>
<start>
<stop>
```

Syntax (MX104 Router)

```
test services rpm rfc2544-benchmarking test
<test-name>
<test-id>
<start>>
<stop>
```

Release Information

Command introduced in Junos OS Release 12.3X52 for ACX Series routers.
Command introduced in Junos OS Release 13.3R1 for MX104 Universal Routing Platforms.

Description

Start or stop an RFC 2544-based benchmarking test. You can start or stop all of the test names that are defined on a router, or start or stop a specific test name. You can also stop a test based on its test identifier. You can also clear the statistical counters associated with the test. When you trigger an RFC 2544-based benchmarking test, it passes through a series of states. These states are displayed in the Test state field in the brief or displayed output information of the **show services rpm rfc2544-benchmarking** command.

NOTE: The RFC 2544 test is stopped at the initiator automatically after the test successfully completes all of the test steps. You need not explicitly enter the **test services rpm rfc2544-benchmarking test <*test-name* | *test-id*> stop** command. However, at the reflector, you must explicitly enter this command to stop the test after the test is completed at the initiator.

When a Layer 2 circuit pseudowire is not up, you cannot start the RFC 2544-based benchmarking test in reflection test mode by entering the **test services rpm rfc2544-benchmarking test *test-name* start** command. If you attempt to start the reflection test mode, a message is displayed explaining the reason for the failure to commence the test.

Options

start—Start the RFC 2544-based benchmarking test

stop—Terminate the RFC 2544-based benchmarking test

clear-counters—(ACX Series routers only) Clear the statistics associated with the benchmarking test that was run.

routing-instance—(ACX Series routers only) (Optional) Name of the routing instance for the test.

test-name—(Optional) Name of the benchmarking test that must be started or stopped.

test-id—(Optional) Unique identifier of the test that must be stopped. You can stop a test based on the test identifier. You can use the ***test-id*** option with only the **test services rpm rfc2544-benchmarking stop** command.

Additional Information

The test session is supported in out-of-service mode for the underlying service. You must not transmit any traffic to the UNI port, configured as a generator or a reflector, that is being tested during the duration of the test.

Required Privilege Level

view

RELATED DOCUMENTATION

Configuring an RFC 2544-Based Benchmarking Test | **659**

Understanding RFC2544-Based Benchmarking Tests on MX Series Routers | **647**

rfc2544-benchmarking | **1108**

List of Sample Output

test services rpm rfc2544-benchmarking test start on page 1441

Output Fields

To display the results of the benchmarking test, use the **show services rpm rfc2544-benchmarking test start** command.

Sample Output

test services rpm rfc2544-benchmarking test start

```
user@host> test services rpm rfc2544-benchmarking test test1 start
```

```
Test "test1" id 56 started
```

The response specifies that a test has been started with test id 56. The test ID can be further used in **show** commands to view test output.