Flow Aggregation

Published: 2014-05-02

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, California 94089
USA
408-745-2000
www.juniper.net

Flow Aggregation

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

List of Tables

About the Documentation

Documentation and Release Notes

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

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

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

Supported Platforms

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

- M Series
- T Series
- MX Series

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

Table 1 on page ix 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 x defines the text and syntax conventions used in this guide.

Table 2: Text and Syntax Conventions

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

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

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

GUI Conventions

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

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

Documentation Feedback

We encourage you to provide feedback, comments, and suggestions so that we can improve the documentation. You can send your comments to techpubs-comments@juniper.net, or fill out the documentation feedback form at https://www.juniper.net/cgi-bin/docbugreport/. If you are using e-mail, be sure to include the following information with your comments:

- Document or topic name
- URL or page number
- Software release version (if applicable)

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/

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

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

Opening a Case with JTAC

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

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

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

PART 1

Overview

CHAPTER 1

Flow Aggregation

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 on page 7
- Configuring Flow Aggregation to Use Version 5 or Version 8 cflowd on page 8
- Configuring Flow Aggregation to Use Version 9 Flow Templates on page 12
- *Directing Replicated Flows to Multiple Flow Servers*
- *Logging cflowd Flows Before Export*

PART 2

Configuration

CHAPTER 2

Configuration Tasks

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 on page 3
- Configuring Flow Aggregation to Use Version 5 or Version 8 cflowd on page 8
- Configuring Flow Aggregation to Use Version 9 Flow Templates on page 12
- *Directing Replicated Flows to Multiple Flow Servers*
- *Configuring Traffic Sampling*
- *Example: Configuring Active Flow Monitoring Version 9 for IPv6*

- *Logging cflowd Flows Before Export*

Configuring Flow Aggregation 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.juniper.net;
   server 172.17.28.5;
}
processes {
   ntp enable;
}
```

For more information, see the *Junos OS Administration Library for Routing Devices*.

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 would be 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*.

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 RE- 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 RE- 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 RE- 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 153.104.248.37 {
         port 9996;
         version 5;
      }
      interface sp-2/2/0 {
      engine-id 4;
      source-address 153.104.0.254;
   }
}
```

The following example shows an RE-based flow aggregation configuration using version 5:

```
family inet {
  output {
    flow-inactive-timeout 15;
    flow-active-timeout 60;
    flow-server 153.104.248.37 {
      port 9996;
      source-address 153.104.0.254;
      version 5;
    }
  }
```

Related Documentation

- Understanding Flow Aggregation on page 3
- Enabling Flow Aggregation on page 7
- Configuring Flow Aggregation to Use Version 9 Flow Templates on page 12
- Configuring Flow Aggregation to Use IPFIX Flow Templates on page 22

Configuring Flow Aggregation to Use Version 9 Flow Templates

Use of version 9 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.

NOTE: Version 9 requires that you install a services PIC, such as the Adaptive Services PIC or Multiservices PIC in the router. On MX Series routers, the Multiservices 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 will originate 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 on page 13
- Configuring the Version 9 Template Properties on page 13
- Customizing Template ID, Observation Domain ID, and Source ID for Version 9 flow Templates on page 14
- Restrictions on page 15

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 the maximum packet length (beyond which the packets are truncated). maximum packets to be sampled per second (beyond which the packets are dropped), the rate (for example, if you specify 10, every 10th packet is sampled), and run length (which specify 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 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 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 *name*** statement.
- You then specify each template for the appropriate type of traffic by including the **ipv4-template**, **ipv6–template**, **mpls-ipv4-template**, **mpls-template**, or **peer-as-billing-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

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. Starting with Junos OS Release 14.1, 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, template IDs can be reassigned. In Junos OS releases earlier than Release 14.1, template IDs and options template IDs were predefined for each address family and could not be modified.

This functionality to configure template ID, options template ID, observation domain ID, and source ID is supported on all routers with MPCs (Trio chip-based FPCs).

The following values were assigned by default for the template IDs of IPFIX templates for the different protocols or address families, until Junos OS Release 13.3:

- IPv4 flow template ID—256
- IPv6 flow template ID—257
- VPLS flow template ID—258
- Options template ID for all address families—512

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 33 and "Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows" on page 28.

Restrictions

The following restrictions apply to version 9 templates:

- You cannot apply the two different types of flow aggregation configuration (cflowd version 5/8 and flow aggregation version 9) 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) would look 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 would be 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

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 *Junos OS MPLS Applications Library for Routing Devices*.

1. 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.

Previously, IPv4 packets (only) would have been sent to the PIC for sampling even though you configured MPLS sampling. No flows should be created, with the result that the parser fails.

With the current capability of applying MPLS templates, MPLS flows are created.

2. 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.

3. 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 is a sample 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 peer-as-billing-template-1 {
        peer-as-billing-template;
        }
```

```
        }
      }
    }
  }
```

The following is a sample firewall filter configuration for MPLS traffic:

```
firewall {
  family mpls {
    filter mpls_sample {
      term default {
        then {
          accept;
          sample;
        }
      }
    }
  }
}
```

The following sample configuration 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 1.2.3.4 {
        port 2055;
```

```
            version9 {
              template mpls-ipv4-template-1;
            }
          }
          interface sp-7/0/0 {
            source-address 1.1.1.1;
          }
          }
        }
      }
    }
```

The following is a sample 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 sample configuration 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 sample configuration 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 1.2.25.5;
        as-path AS-2;
      }
      then destination-class dcu2;
  }
  policy-statement P3 {
      from {
        protocol bgp;
        neighbor 192.2.1.1;
        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 peer-as-billing 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 2.3.4.5;
          }
        }
      }
    }
  }
  family inet {
    filter {
      output peer-as-filter;
    }
  }
```

Related Documentation

- Understanding Flow Aggregation on page 3
- Enabling Flow Aggregation on page 7
- Configuring Flow Aggregation to Use Version 5 or Version 8 cflowd on page 8
- Configuring Flow Aggregation to Use IPFIX Flow Templates on page 22
- *Configuring Traffic Sampling*
- *Example: Configuring Active Flow Monitoring Version 9 for IPv6*

Configuring Flow Aggregation to Use IPFIX Flow Templates

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 need not be 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 on page 22
- Restrictions on page 23
- Customizing Template ID, Observation Domain ID, and Source ID for IPFIX flow Templates on page 24
- Fields Included in the IPv4 Template on page 24
- Fields Included in the IPv6 Template on page 25
- Verification on page 26
- Example: Configuring an IPFIX Flow Templates and Flow Sampling on page 26

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.

- VLAN ID field is not valid for egress traffic, and returns a value of *0* for egress traffic.
- 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

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. Starting with Junos OS Release 14.1, 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, template IDs can be reassigned. In Junos OS releases earlier than Release 14.1, template IDs and options template IDs were predefined for each address family and could not be modified.

This functionality to configure template ID, options template ID, observation domain ID, and source ID is supported on all routers with MPCs (Trio chip-based FPCs).

The following values were assigned by default for the template IDs of version 9 templates for the different protocols or address families, until Junos OS Release 13.3:

- IPv4 flow template ID—272
- IPv6 flow template ID—273
- VPLS flow template ID—274
- Options template ID for all address families—520

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 33 and "Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows" on page 28.

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
- 802.1Q VLAN identifier (dot1qVlanId)
- 802.1Q Customer VLAN identifier (dot1qCustomerVlanId)

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
- 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
- 802.1Q VLAN identifier (dot1qVlanId)
- 802.1Q Customer VLAN identifier (dot1qCustomerVlanId)

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 Templates 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

- Understanding Flow Aggregation on page 3
- Enabling Flow Aggregation on page 7
- Configuring Flow Aggregation to Use Version 5 or Version 8 cflowd on page 8
- Configuring Flow Aggregation to Use Version 9 Flow Templates on page 12

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;
}
```

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 3 on page 29 describes the values of data template and option template IDs for different protocols with default and configured values for IPFIX flows.

Table 3: 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

Table 3: Values of Template and Option Template IDs for IPFIX Flows *(continued)*

Family	Configured Value	Data Template	Option Template
VPLS	1024-65535	1024-65535	1024-65535
MPLS	None	259	579
MPLS	1024-65535	1024-65535	1024-65535

Table 4 on page 30 describes the values of data template and option template IDs for different protocols with default and configured values for version 0 flows.

Table 4: 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 3 on page 29 describes the values of data template and option template IDs for different protocols with default and configured values for IPFIX flows.

Table 5: 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
4	IPv4 (0)	2	1	1	0000 0100 0000 1000 0000 0010 0001 0001 0x04080211

Table 5: 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	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

Related Documentation

- Configuring Observation Domain ID and Source ID for Version 9 and IPFIX Flows on page 33

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.

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 6 on page 34 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 6: 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
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

Table 6: 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
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
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

Related Documentation

- Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows on page 28

CHAPTER 3

Configuration Statements

[edit services flow-monitoring] Hierarchy Level

```
services {
  flow-monitoring {
    version9 {
      template template-name {
        flow-active-timeout seconds;
        flow-inactive-timeout seconds;
        ipv4-template {
          nexthop-options {
            mpls {
              label-position [ positions ];
            }
          }
        }
        ipv6-template;
        mpls-template {
          label-position [ positions ];
        }
        mpls-ipv4-template {
          label-position [ positions ];
        }
        option-refresh-rate {
          packets packets;
          seconds seconds;
        }
        peer-as-billing-template;
        template-refresh-rate {
          packets packets;
          seconds seconds;
        }
        peer-as-billing-template;
        option-refresh-rate packets;
        template-refresh-rate packets;
      }
    }
  }
}
```

Related Documentation

- *Notational Conventions Used in Junos OS Configuration Hierarchies*
- *[edit services] Hierarchy Level*

[edit interfaces] Hierarchy Level

To configure flow monitoring and accounting interfaces, include the following statements at the **[edit interfaces]** hierarchy level:

```
[edit interfaces]
mo-fpc/pic/port {
  unit logical-unit-number {
    family inet {
      accounting {
```

```
            destination-class-usage;
            source-class-usage direction;
          }
        }
        address address {
          destination address;
        }
        filter {
          group filter-group-number;
          input filter-name;
          output filter-name;
        }
        receive-options-packets;
        receive-ttl-exceeded;
        sampling direction;
      }
    }
    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;
      }
    }
    (at-fpc/pic/port | fe-fpc/pic/port | ge-fpc/pic/port) {
      passive-monitor-mode;
    }
    so-fpc/pic/port {
      unit logical-unit-number {
        passive-monitor-mode;
      }
    }
```

Related Documentation

- [edit forwarding-options] Hierarchy Level on page 39
- [edit services flow-monitoring] Hierarchy Level on page 38

[edit forwarding-options] Hierarchy Level

To configure flow monitoring and accounting properties, include the following statements at the **[edit forwarding-options]** hierarchy level:

```
[edit forwarding-options]
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;
        }
      }
    }
    monitoring name {
      family family {
        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;
          }
        }
      }
    next-hop-group group-names {
      interface interface-name {
        next-hop address;
      }
    }
    port-mirroring {
      input {
        rate rate;
        run-length number;
        maximum-packet-length bytes
      }
      family (inet | inet6) {
        output {
          interface interface-name {
            next-hop address;
          }
          no-filter-check;
        }
      }
      traceoptions {
        file filename {
```

```
            files number;
            size bytes;
            (world-readable | no-world-readable);
        }
    }
}
sampling {
    disable;
    sample-once;
    input {
        rate number;
        run-length number;
        max-packets-per-second number;
        maximum-packet-length bytes;
    }
    traceoptions {
        no-remote-trace;
        file filename <files number> <size bytes> <match expression> <world-readable |
            no-world-readable>;
    }
    family (inet | inet6 | mpls) {
        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);
                (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);
          }
        }
      }
      instance instance-name {
        disable;
        input {
          rate number;
          run-length number;
          max-packets-per-second number;
          maximum-packet-length bytes;
        }
        family (inet | inet6 | mpls) {
          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);
              (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;
            }
            inline-jflow {
              source-address address;
              flow-export-rate rate;
            }
          }
        }
      }
    }
```

NOTE: For the complete **[edit forwarding-options]** hierarchy, see the *Routing Policy Feature Guide for Routing Devices*. This section documents only the statements used in flow monitoring and accounting services.

Related Documentation

- [edit interfaces] Hierarchy Level on page 38
- [edit services flow-monitoring] Hierarchy Level on page 38

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 on page 7

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.

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 on page 7

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 (Forwarding Options) 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***—The IP address or identifier of the host system (the workstation running the cflowd utility).

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

- Enabling Flow Aggregation on page 7

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 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 Monitoring*

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) output],
[edit forwarding-options sampling family (inet |inet6 |mpls) output],
[edit services flow-monitoring version9]
[edit services flow-monitoringversion-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.

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 f**orwarding-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*
- Configuring the Version 9 Template Properties on page 13

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) output],
[edit forwarding-options sampling family (inet |inet6 |mpls) output],
[edit services flow-monitoring version9]
[edit services flow-monitoringversion-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.

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: 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)

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*
- Configuring the Version 9 Template Properties on page 13

ipv4-template

Syntax ipv4-template;

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]
[edit services flow-monitoringversion-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.

Description Specify that the flow aggregation version 9 template is used only for IPv4 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 to Use Version 9 Flow Templates on page 12

ipv6-template

Syntax ipv6-template;

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]
[edit services flow-monitoringversion-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.

Description Specify that the flow aggregation version 9 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 to Use Version 9 Flow Templates on page 12

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 to Use Version 9 Flow Templates on page 12

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 on page 7

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: 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*

mpls-ipv4-template

Syntax

```
mpls-ipv4-template {
  label-position [ positions ];
}
```

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]

Release Information Statement introduced in Junos OS Release 8.3.

Description Specify the flow aggregation version 9 properties for templates that combine IPv4 and MPLS records. The remaining statement is 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 to Use Version 9 Flow Templates on page 12

mpls-template

Syntax

```
mpls-template {
  label-position [ positions ];
}
```

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]

Release Information Statement introduced in Junos OS Release 8.3.

Description Specify the flow aggregation version 9 properties for templates used only for MPLS records. The remaining statement is 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 to Use Version 9 Flow Templates on page 12

observation-domain-id

Syntax observation-domain-id *domain-id*;

Hierarchy Level [edit services flow-monitoringversion-ipfix template *template-name*]

Release Information Statement introduced in Junos OS Release 14.1.

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 inquie 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***—Specify a 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 on page 33
- Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows on page 28

option-refresh-rate

Syntax option-refresh-rate packets *packets* seconds *seconds*;

Hierarchy Level [edit services flow-monitoring version9],
[edit services flow-monitoring version9 template *template-name*]
[edit services flow-monitoringversion-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.

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

- Configuring Flow Aggregation to Use Version 9 Flow Templates on page 12

options-template-id

Syntax options-template-id *id*;

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]
[edit services flow-monitoringversion-ipfix template *template-name*]

Release Information Statement introduced in Junos OS Release 14.1.

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*—Specify a unique identifier for the options template to be used for version 9 or IPFIX flows.

Range: 1024 through 65535

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 on page 33
- Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows on page 28

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 to Use Version 9 Flow Templates on page 12

port

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 on page 7

rate (Forwarding Options)

Syntax rate *number*;

Hierarchy Level [edit forwarding-options port-mirroring input],
[edit forwarding-options sampling input],
[edit forwarding-options sampling instance *instance-name* input],
[edit forwarding-options port-mirroring family (inet|inet6) 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.

Description Set a 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.

Options ***number***—Denominator of the ratio.

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
- *Configuring Port Mirroring*
- *Configuring Traffic Sampling*

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.

Description Set the number of samples following the initial trigger event. The configuration enables you to sample packets following those already being sampled.

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 Filters to Forwarding Tables*
- *Configuring Port Mirroring*
- *Configuring Traffic Sampling*

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***—Specify a 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 on page 33
- Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows on page 28

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 to Use Version 9 Flow Templates on page 12

template-id

Syntax template-id *id*;

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]
[edit services flow-monitoringversion-ipfix template *template-name*]

Release Information Statement introduced in Junos OS Release 14.1.

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*—Specify a unique identifier for the template to be used for version 9 or IPFIX flows.

Range: 1024 through 65535

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 on page 33
- Configuring Template ID and Options Template ID for Version 9 and IPFIX Flows on page 28

template-refresh-rate

Syntax template-refresh-rate packets *packets* seconds *seconds*;

Hierarchy Level [edit services flow-monitoring version9 template *template-name*]
[edit services flow-monitoringversion-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.

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

- Configuring Flow Aggregation to Use Version 9 Flow Templates on page 12

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*
- Enabling Flow Aggregation on page 7

PART 3

Index

Index

T

V