JUNIPER NETWORKS

Junos® OS

CoS Forwarding Classes Feature Guide for Routing Devices

Release
13.2

Published: 2013-07-31

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

Junos® OS CoS Forwarding Classes Feature Guide for Routing Devices
13.2

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

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

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

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

Supported Platforms

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

- T Series
- M Series
- MX Series
- J Series
- PTX 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 xiii 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.

Table 2 on page xiii 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`

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

Convention	Description	Examples
Italic text like this	• Introduces or emphasizes important new terms. • Identifies book names. • Identifies RFC and Internet draft titles.	• A policy *term* is a named structure that defines match conditions and actions. • *Junos OS System Basics Configuration 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)	Enclose 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)	Enclose a variable for which you can substitute one or more values.	**community name members [** ***community-ids*** **]**

Indention and braces ({ })	Identify 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		
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: https://www.juniper.net/alerts/

- 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

Forwarding Classes

Overview of Forwarding Classes

This topic covers the following information:

Output Queue Assignments Based on Forwarding Class

It is helpful to think of forwarding classes as output queues. In effect, the end result of classification is the identification of an output queue for a particular packet.

CoS packet classification assigns an incoming packet to an output queue based on the packet's forwarding class. Each packet is associated with one of the following default forwarding classes:

- Expedited forwarding (EF)—Provides a low-loss, low-latency, low- jitter, assured bandwidth, end-to-end service.
- Assured forwarding (AF)—Provides a group of values you can define and includes four subclasses: AF1, AF2, AF3, and AF4, each with three drop probabilities: low, medium, and high.
- Best effort (BE)—Provides no service profile. For the best effort forwarding class, loss priority is typically not carried in a class-of-service (CoS) value and random early detection (RED) drop profiles are more aggressive.
- Network control (NC)—This class is typically high priority because it supports protocol control.

Devices That Support Up to Four Forwarding Classes

Some of the Juniper Networks routing platforms support up to four forwarding classes for classifying customer traffic. On these platforms, you can configure one of each type of default forwarding class. The following Juniper Networks routing platforms support up to four forwarding classes:

- M7i Multiservice Edge Routers with Compact Forwarding Engine Boards (CFEBs)
- M10i Multiservice Edge Routers with CFEBs

NOTE: This list does not reference any Juniper Networks device that has reached its End of Life (EOL) period and its End of Support (EOS) milestone date.

Devices That Support Up to 16 Forwarding Classes

Other Juniper Networks routing platforms support up to 16 forwarding classes, which enables you to classify packets more granularly. For example, you can configure multiple classes of EF traffic: EF, EF1, and EF2. On these platforms, the Junos OS software supports up to eight output queues; therefore, if you configure more than eight forwarding classes, you must map multiple forwarding classes to single output queues. The following Juniper Networks routing and switching platforms support up to 16 forwarding classes and up to 8 output queues:

- EX Series switches
- M7i Multiservices Edge Routers with Enhanced Compact Forwarding Engine Boards (CFEB-Es)
- M10i Multiservices Edge Routers with CFEB-Es
- M120 Multiservices Edge Routers
- M320 Multiservices Edge Routers
- MX Series 3D Universal Edge Routers
- T Series Core Routers
- PTX Packet Transport Routers

Default and Configurable Packet Loss Priority Values

By default, the loss priority is low. On most devices, you can configure high or low loss priority. On the following devices, you can configure high, low, medium-high, or medium-low loss priority:

- J Series Services Router interfaces
- M320 routers and T Series routers with Enhanced II Flexible PIC Concentrators (FPCs)

- T640 routers with Enhanced Scaling FPC4s
- PTX Series Packet Transport Routers

Configuration Statements Used to Configure and Apply Forwarding Classes

To configure CoS forwarding classes, include the **forwarding-classes** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-classes {
   class class-name queue-num queue-number priority (high | low);
   queue queue-number class-name priority (high | low);
}
forwarding-class-map forwarding-class-map-name {
   class class-name queue-num queue-number [ restricted-queue queue-number ];
}
interfaces {
   interface-name {
      unit logical-unit-number {
         forwarding-class class-name;
         forwarding-class-map forwarding-class-map-name;
      }
   }
}
restricted-queues {
   forwarding-class class-name queue queue-number;
}
```

Related Documentation

- Default Forwarding Classes on page 5
- Configuring Forwarding Classes on page 19
- Applying Forwarding Classes to Interfaces on page 20
- Configuring Up to 16 Forwarding Classes on page 24
- *Policer Overview*

Default Forwarding Classes

By default, four queues are assigned to four forwarding classes, each with a queue number, name, and abbreviation.

These default mappings apply to all routers. The four forwarding classes defined by default are shown in Table 3 on page 6.

If desired, you can rename the forwarding classes associated with the queues supported on your hardware. Assigning a new class name to an output queue does not alter the default classification or scheduling that is applicable to that queue. CoS configurations can be quite complicated, so unless it is required by your scenario, we recommend that you not alter the default class names or queue number associations.

Some routers support eight queues. Queues 4 through 7 have no default mappings to forwarding classes. To use queues 4 through 7, you must create custom forwarding class

names and map them to the queues. For more information, see the Juniper Networks J Series Services Router documentation.

Table 3: Default Forwarding Classes

Queue	Forwarding Class Name	Comments
Queue 0	**best-effort** (**be**)	The software does not apply any special CoS handling to packets with 000000 in the DiffServ field, a backward compatibility feature. These packets are usually dropped under congested network conditions.
Queue 1	**expedited-forwarding** (**ef**)	The software delivers assured bandwidth, low loss, low delay, and low delay variation (jitter) end-to-end for packets in this service class. Routers accept excess traffic in this class, but in contrast to assured forwarding, out-of-profile expedited-forwarding packets can be forwarded out of sequence or dropped.
Queue 2	**assured-forwarding** (**af**)	The software offers a high level of assurance that the packets are delivered as long as the packet flow from the customer stays within a certain service profile that you define. The software accepts excess traffic, but applies a RED drop profile to determine if the excess packets are dropped and not forwarded. Depending on router type, up to four drop probabilities (low, medium-low, medium-high, and high) are defined for this service class.
Queue 3	**network-control** (**nc**)	The software delivers packets in this service class with a low priority. (These packets are not delay sensitive.) Typically, these packets represent routing protocol hello or keepalive messages. Because loss of these packets jeopardizes proper network operation, delay is preferable to discard.

The following rules govern queue assignment:

- If classifiers fail to classify a packet, the packet always receives the default classification to the class associated with queue 0.
- The number of queues is dependent on the hardware plugged into the chassis. CoS configurations are inherently contingent on the number of queues on the system. Only two classes, **best-effort** and **network-control**, are referenced in the default configuration. The default configuration works on all routers.
- CoS configurations that specify more queues than the router can support are not accepted. The commit fails with a detailed message that states the total number of queues available.
- All default CoS configuration is based on queue number. The name of the forwarding class that shows up when the default configuration is displayed is the forwarding class currently associated with that queue.

This is the default configuration for the **forwarding-classes** statement:

```
[edit class-of-service]
```

```
forwarding-classes {
  queue 0 best-effort;
  queue 1 expedited-forwarding;
  queue 2 assured-forwarding;
  queue 3 network-control;
}
```

If you reassign the forwarding-class names, the **best-effort** forwarding-class name appears in the locations in the configuration previously occupied by **network-control** as follows:

```
[edit class-of-service]
forwarding-classes {
  queue 0 network-control;
  queue 1 assured-forwarding;
  queue 2 expedited-forwarding;
  queue 3 best-effort;
}
```

All the default rules of classification and scheduling that applied to Queue 3 still apply. Queue 3 is simply now renamed **best-effort**.

On Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers, you can assign multiple forwarding classes to a single queue. If you do so, the first forwarding class that you assign to queue 0 acquires the default BE classification and scheduling. The first forwarding class that you assign to queue 1 acquires the default EF classification and scheduling. The first forwarding class that you assign to queue 2 acquires the default AF classification and scheduling. The first forwarding class that you assign to queue 3 acquires the default NC classification and scheduling. For more information, see "Configuring Up to 16 Forwarding Classes" on page 24.

- In the current default configuration:
  - Only IP precedence classifiers are associated with interfaces.
  - The only classes designated are **best-effort** and **network-control**.
  - Schedulers are not defined for the **expedited-forwarding** or **assured-forwarding** forwarding classes.
- You must explicitly classify packets to the **expedited-forwarding** or **assured-forwarding** forwarding class and define schedulers for these classes.
- For Asynchronous Transfer Mode (ATM) interfaces on Juniper Networks M Series Multiservice Edge Routers, when you use fixed classification with multiple logical interfaces classifying to separate queues, a logical interface without a classifier attached inherits the most recent classifier applied on a different logical interface. For example, suppose you configure traffic through logical unit 0 to be classified into queue 1, and you configure traffic through logical unit 1 to be classified into queue 3. You want traffic through logical unit 2 to be classified into the default classifier, which is queue 0. In this case, traffic through logical unit 2 is classified into queue 3, because the configuration of logical unit 1 was committed last.

Related Documentation

- Overview of Forwarding Classes on page 3
- Configuring Forwarding Classes on page 19
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13
- *CoS Features and Limitations on J, M, and T Series Routers*
- *CoS Features and Limitations on PTX Series Packet Transport Routers*

Forwarding Classes and Fabric Priority Queues

This topic covers the following information:

Default Fabric Priority Queuing

On Juniper Networks EX Series switches, M320 Multiservice Edge Routers, and Juniper Networks T Series Core Routers, the default behavior is for fabric priority queuing on egress interfaces to match the scheduling priority you assign. High-priority egress traffic is automatically assigned to high-priority fabric queues. Likewise, low-priority egress traffic is automatically assigned to low-priority fabric queues.

Overriding Default Fabric Priority Queuing

You can override the default fabric priority queuing of egress traffic by including the **priority** statement at the **[edit class-of-service forwarding-classes queue *queue-number class-name*]** hierarchy level:

```
[edit class-of-service forwarding-classes queue queue-number class-name]
priority (high | low);
```

Related Documentation

- *Associating Schedulers with Fabric Priorities*

CHAPTER 2

Host Outbound Traffic

Understanding Queuing and Marking of Host Outbound Traffic

This topic covers the following information:

Host Outbound Traffic Overview

Host outbound traffic, also called locally generated traffic, consists of traffic generated by the Routing Engine and traffic generated by the distributed protocol handler.

Routing Engine Sourced Traffic

Traffic sent from the Routing Engine includes control plane packets such as OSPF Hello packets, ICMP echo reply (ping) packets, and TCP-related packets such as BGP and LDP control packets.

Distributed Protocol Handler Traffic

Distributed protocol handler traffic refers to traffic from the router's *periodic packet management* (PPM) process when it runs sessions distributed to the Packet Forwarding Engine (the default mode) in addition to the Routing Engine. The PPM process is responsible for periodic transmission of protocol Hello or other keepalive packets on behalf of its various client processes, such as Bidirectional Forwarding Detection (BFD) Protocol or Link Aggregation Control Protocol (LACP), and it also receives packets on behalf of client processes. In addition, PPM handles time-sensitive periodic processing and performs such tasks as sending process-specific packets and gathering statistics. By default, PPM sessions on the Routing Engine run distributed on the Packet Forwarding Engine, and this enables client processes to run on the Packet Forwarding Engine.

NOTE: For interfaces on MX80 routers, LACP control traffic is sent through the Routing Engine rather than through the Packet Forwarding Engine.

Distributed protocol handler traffic includes both IP (Layer 3) traffic such as BFD keepalivemessages and non-IP (Layer 2) traffic such as LACP control traffic on aggregated Ethernet.

Default Queuing and Marking of Host Outbound Traffic

By default, the router assigns host outbound traffic to the **best-effort** forwarding class (which maps to queue 0) or to the **network-control** forwarding class (which maps to queue 3) based on protocol. For more information, see "Default Queue Assignments for Routing Engine Sourced Traffic" on page 11.

By default, the router marks the type of service (ToS) field of Layer 3 packets in the host outbound traffic flow with DifffServ code point (DSCP) bits 000000 (which correlate with the **best-effort** forwarding class). The router does not remark Layer 2 traffic such as LACP control traffic on aggregated Ethernet. For more information, see *Default DSCP and DSCP IPv6 Classifier*.

Configured Queuing and Marking of Host Outbound Traffic

You can configure a nondefault forwarding class and DSCP bits that the router uses to queue and remark host outbound traffic. These configuration settings apply to the following types of traffic:

- Packets generated by the Routing Engine
- Distributed protocol handler traffic for egress interfaces hosted on MX Series routers, M120 routers, and Enhanced III FPCs in M320 routers.

To change these default settings, include the **forwarding-class *class-name*** statement and the **dscp-code-point *value*** statement at the **[edit class-of-service host-outbound-traffic]** hierarchy level. This feature does not affect transit traffic or incoming traffic.

The configured forwarding class override applies to all packets relating to Layer 2 protocols, Layer 3 protocols, and all application-level traffic (such as FTP or ping operations). The configured DSCP bits override value does not apply to MPLS EXP bits or IEEE 802.1p bits, however.

Configured Queuing and Marking of Outbound Routing Engine Traffic Only

To configure a nondefault forwarding class and DSCP bits that the router uses to queue and remark traffic generated by the Routing Engine only, attach an IPv4 firewall filter to the output of the router's loopback address. Use the **forwarding-class** and **dscp** filter actions to specify override values.

This feature overrides the **host-outbound-traffic** settings for the Routing Engine output traffic only.

Related Documentation

- Default Queue Assignments for Routing Engine Sourced Traffic on page 11
- *Default DSCP and DSCP IPv6 Classifier*
- Example: Configuring Different Queuing and Marking Defaults for Outbound Routing Engine and Distributed Protocol Handler Traffic on page 31

Default Queue Assignments for Routing Engine Sourced Traffic

Table 4 on page 11 lists (in alphabetical order) how Routing Engine sourced traffic is mapped by protocol type to output queues. The follow caveats apply to this information:

- For all packets sent to queue 3 over a VLAN-tagged interface, the software sets the 802.1p bits to 110.
- For IPv4 and IPv6 packets, the software copies the IP type-of-service (ToS) value into the 802.1p field independently of the queue from which the packets are sent .
- For MPLS packets, the software copies the EXP bits into the 802.1p field.

Table 4: Default Queue Assignments for Packets Generated by the Routing Engine

Routing Engine Protocol	Default Queue Assignment
Adaptive Services PIC TCP tickle (keepalive packets for idle session generated with stateful firewall to probe idle TCP sessions)	Queue 0
ATM Operation, Administration, and Maintenance (OAM)	Queue 3
Bidirectional Forwarding Detection (BFD) Protocol	Queue 3
BGP	Queue 0
BGP TCP Retransmission	Queue 3
Cisco High-Level Data Link Control (HDLC)	Queue 3
Distance Vector Multicast Routing Protocol (DVMRP)	Queue 3
Frame Relay Local Management Interface (LMI)	Queue 3
Frame Relay Asynchronization permanent virtual circuit (PVC)/data link connection identifier (DLCI) status messages	Queue 3
FTP	Queue 0
IS-IS Open Systems Interconnection (OSI)	Queue 3
Internet Group Management Protocol (IGMP) query	Queue 3
IGMP Report	Queue 0

Table 4: Default Queue Assignments for Packets Generated by the Routing Engine *(continued)*

Routing Engine Protocol	Default Queue Assignment
IP version 6 (IPv6) Neighbor Solicitation	Queue 3
IPv6 Neighbor Advertisement	Queue 3
IPv6 Router Advertisement	Queue 0
Label Distribution Protocol (LDP) User Datagram Protocol (UDP) hello	Queue 3
LDP keepalive and Session data	Queue 0
LDP TCP Retransmission	Queue 3
Link Aggregation Control Protocol (LACP)	Queue 3
Link Services (LS) PIC	If link fragmentation and interleaving (LFI) is enabled, all routing protocol packets larger than 128 bytes are transmitted from queue 0. This ensures that VoIP traffic is not affected. Fragmentation is supported on queue 0 only.
Multicast listener discovery (MLD)	Queue 0
Multicast Source Discovery Protocol (MSDP)	Queue 0
MSDP TCP Retransmission	Queue 3
Multilink Frame Relay Link Integrity Protocol (LIP)	Queue 3
OSPF protocol data unit (PDU)	Queue 3
Point-to-Point Protocol (PPP)	Queue 3
Protocol Independent Multicast (PIM)	Queue 3
Real-time performance monitoring (RPM) probe packets	Queue 3
RSVP	Queue 3
Routing Information Protocol (RIP)	Queue 3
SNMP	Queue 0
SSH	Queue 0
Telnet	Queue 0
Virtual Router Redundancy Protocol (VRRP)	Queue 3

Table 4: Default Queue Assignments for Packets Generated by the Routing Engine *(continued)*

Routing Engine Protocol	Default Queue Assignment
xnm-clear-text	Queue 0
xnm-ssl	Queue 0

Related Documentation

- Overview of Forwarding Classes on page 3
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13

Changing the Default Queuing and Marking of Host Outbound Traffic

You can modify the default queue assignment (forwarding class) and DSCP bits used in the ToS field of *host outbound traffic* (packets generated by the Routing Engine).

TCP-related packets, such as BGP or LDP, use queue 3 (network control) for retransmitted traffic. Changing the defaults for Routing Engine sourced traffic does not affect transit or incoming traffic. The changes apply to all packets relating to Layer 3 and Layer 2 protocols, but not MPLS EXP bits or IEEE 802.1p bits. This feature applies to all application-level traffic such as FTP or ping operations as well.

This feature is not available on Juniper Networks J Series Services Routers.

The queue selected is global to the routing device. That is, the traffic is placed in the selected queue on all egress interfaces. In the case of a restricted interface, the Routing Engine sourced traffic flows through the restricted queue.

The queue selected must be properly configured on all interfaces.

To change the default queue and DSCP bits for Routing Engine sourced traffic, include the **host-outbound-traffic** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
host-outbound-traffic {
  forwarding-class class-name;
  dscp-code-point value;
}
```

The following example places all Routing Engine sourced traffic into queue 3 (network control) with a DSCP value of 101010:

```
[edit class-of-service]
host-outbound-traffic {
  forwarding-class network-control;
  dscp-code-point 101010;
}
```

Related Documentation

- Overview of Forwarding Classes on page 3
- Default Queue Assignments for Routing Engine Sourced Traffic on page 11

- *Default DSCP and DSCP IPv6 Classifier*

CHAPTER 3

CoS-Based Forwarding

Forwarding Policy Options Overview

Class-of-service (CoS)-based forwarding (CBF) enables you to control next-hop selection based on a packet's class of service and, in particular, the value of the IP packet's precedence bits.

For example, you might want to specify a particular interface or next hop to carry high-priority traffic while all best-effort traffic takes some other path. When a routing protocol discovers equal-cost paths, it can pick a path at random or load-balance across the paths through either hash selection or round robin. CBF allows path selection based on class.

To configure CBF properties, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-policy {
  next-hop-map map-name {
    forwarding-class class-name {
      next-hop [ next-hop-name ];
      lsp-next-hop [ lsp-regular-expression ];
      non-lsp-next-hop;
      discard;
    }
  }
  class class-name {
    classification-override {
      forwarding-class class-name;
    }
  }
}
```

PART 2

Configuration

CHAPTER 4

Forwarding Class Configuration

Configuring Forwarding Classes

You assign each forwarding class to an internal queue number by including the **forwarding-classes** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-classes {
  class queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low) [ policing-priority (premium |
      normal) ];
  }
```

You cannot commit a configuration that assigns the same forwarding class to two different queues.

CAUTION: We do not recommend classifying packets into a forwarding class that has no associated scheduler on the egress interface. Such a configuration can cause unnecessary packet drops because an unconfigured scheduling class might lack adequate buffer space. For example, if you configure a custom scheduler map that does not define queue 0, and the default classifier assigns incoming packets to the best-effort class (queue 0), the unconfigured egress queue for the best-effort forwarding class might not have enough space to accommodate even short packet bursts.

A default congestion and transmission control mechanism is used when an output interface is not configured for a certain forwarding class, but receives packets destined for that unconfigured forwarding class. This default

mechanism uses the delay buffer and weighted round robin (WRR) credit allocated to the designated forwarding class, with a default drop profile. Because the buffer and WRR credit allocation is minimal, packets might be lost if a larger number of packets are forwarded without configuring the forwarding class for the interface.

Related Documentation

- Overview of Forwarding Classes on page 3
- Default Forwarding Classes on page 5
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13

Applying Forwarding Classes to Interfaces

You can configure *fixed classification* on a logical interface by specifying a forwarding class to be applied to all packets received by the logical interface, regardless of the packet contents.

NOTE: On the T4000 router, BA classification and fixed classification are mutually exclusive. That is, only one of the classifications can be configured.

To apply a forwarding class configuration to the input logical interface, include the **forwarding-class** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
forwarding-class class-name;
```

You can include interface wildcards for ***interface-name*** and ***logical-unit-number***.

In the following example, all packets coming into the router from the **ge-3/0/0.0** interface are assigned to the **assured-forwarding** forwarding class:

```
[edit class-of-service]
interfaces {
  ge-3/0/0 {
    unit 0 {
      forwarding-class assured-forwarding;
    }
  }
}
```

Related Documentation

- forwarding-class on page 53

Classifying Packets by Egress Interface

For Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers with the Intelligent Queuing (IQ), IQ2, Enhanced IQ (IQE), Multiservices link services intelligent

queuing (LSQ) interfaces, or ATM2 PICs, you can classify unicast and multicast packets based on the egress interface. For unicast traffic, you can also use a multifield filter, but only egress interface classification applies to multicast traffic as well as unicast traffic. If you configure egress classification of an interface, you cannot perform Differentiated Services code point (DSCP) rewrites on the interface. By default, the system will not perform any classification based on the egress interface.

To enable packet classification by the egress interface, you first configure a forwarding class map and one or more queue numbers for the egress interface at the **[edit class-of-service forwarding-class-map *forwarding-class-map-name*]** hierarchy level:

```
[edit class-of-service]
forwarding-class-map  forwarding-class-map-name {
   class class-name queue-num queue-number [ restricted-queue queue-number ];
}
```

For T Series routers that are restricted to only four queues, you can control the queue assignment with the **restricted-queue** option, or you can allow the system to automatically determine the queue in a modular fashion. For example, a map assigning packets to queue 6 would map to queue 2 on a four-queue system.

NOTE: If you configure an output forwarding class map associating a forwarding class with a queue number, this map is not supported on multiservices link services intelligent queuing (lsq-) interfaces.

Once the forwarding class map has been configured, you apply the map to the logical interface by using the **output-forwarding-class-map** statement at the **[edit class-of-service interfaces *interface-name* unit *logical-unit-number*]** hierarchy level:

```
[edit class-of-service interfaces interface-name unit logical-unit-number]
output-forwarding-class-map forwarding-class-map-name;
```

All parameters relating to the queues and forwarding class must be configured as well. For more information about configuring forwarding classes and queues, see "Configuring Forwarding Classes" on page 19.

This example shows how to configure an interface-specific forwarding-class map named **FCMAP1** that restricts queues 5 and 6 to different queues on four-queue systems and then applies **FCMAP1** to **unit 0** of interface **ge-6/0/0**:

```
[edit class-of-service]
forwarding-class-map FCMAP1 {
   class FC1 queue-num 6 restricted-queue 3;
   class FC2 queue-num 5 restricted-queue 2;
   class FC3 queue-num 3;
   class FC4 queue-num 0;
   class FC3 queue-num 0;
   class FC4 queue-num 1;
}

[edit class-of-service]
interfaces {
   ge-6/0/0 unit 0 {
```

```
      output-forwarding-class-map FCMAP1;
    }
  }
```

Note that without the **restricted-queue** option in **FCMAP1**, the example would assign **FC1** and **FC2** to queues 2 and 1, respectively, on a system restricted to four queues.

Use the **show class-of-service forwarding-class** ***forwarding-class-map-name*** command to display the forwarding-class map queue configuration:

```
user@host> show class-of-service forwarding-class FCMAP2

Forwarding class                          ID      Queue  Restricted queue
  FC1                                      0       6          3
  FC2                                      1       5          2
  FC3                                      2       3          3
  FC4                                      3       0          0
  FC5                                      4       0          0
  FC6                                      5       1          1
  FC7                                      6       6          2
  FC8                                      7       7          3
```

Use the **show class-of-service interface** ***interface-name*** command to display the forwarding-class maps (and other information) assigned to a logical interface:

```
user@host> show class-of-service interface ge-6/0/0

Physical interface: ge-6/0/0, Index: 128
Queues supported: 8, Queues in use: 8
 Scheduler map: <default>, Index: 2
 Input scheduler map: <default>, Index: 3
 Chassis scheduler map: <default-chassis>, Index: 4

Logical interface: ge-6/0/0.0, Index: 67
 Object                    Name              Type                    Index
 Scheduler-map             sch-map1          Output                   6998
 Scheduler-map             sch-map1          Input                    6998
 Classifier                dot1p             ieee8021p                4906
 forwarding-class-map      FCMAP1            Output                   1221

Logical interface: ge-6/0/0.1, Index 68
 Object                    Name              Type                    Index
 Scheduler-map             <default>         Output                     2
 Scheduler-map             <default>         Input                      3

Logical interface: ge-6/0/0.32767, Index 69
 Object                    Name              Type                    Index
 Scheduler-map             <default>         Output                     2
 Scheduler-map             <default>         Input                      3
```

Assigning Forwarding Class and DSCP Value for Routing Engine-Generated Traffic

You can set the forwarding class and differentiated service code point (DSCP) value for traffic originating in the Routing Engine. To configure forwarding class and DSCP values that apply to Routing Engine–generated traffic only, apply an output filter to the loopback (**lo.0**) interface and set the appropriate forwarding class and DSCP bit configuration for various protocols. For example, you can set the DSCP value on OSPF packets that originate

in the Routing Engine to **10** and assign them to the AF (assured forwarding) forwarding class while the DSCP value on ping packets are set to **0** and use forwarding class BE (best effort).

This particular classification ability applies to packets generated by the Routing Engine only.

The following example assigns Routing Engine sourced ping packets (using ICMP) a DSCP value of **38** and a forwarding class of **af17**, OSPF packets a DSCP value of **12** and a forwarding class of **af11**, and BGP packets (using TCP) a DSCP value of **10** and a forwarding class of **af16**.

```
[edit class-of-service]
forwarding-classes {
   class af11 queue-num 7;
   class af12 queue-num 1;
   class af13 queue-num 2;
   class af14 queue-num 4;
   class af15 queue-num 5;
   class af16 queue-num 4;
   class af17 queue-num 6;
   class af18 queue-num 7;
}


[edit firewall filter family inet]
filter loopback-filter {
   term t1 {
      from {
         protocol icmp; # For pings
      }
      then {
         forwarding-class af17;
         dscp 38;
      }
   }
   term t2 {
      from {
         protocol ospf; # For OSPF
      }
      then {
         forwarding-class af11;
         dscp 12;
      }
   }
   term t3 {
      from {
         protocol tcp; # For BGP
      }
      then {
         forwarding-class af16;
         dscp 10;
      }
   }
   term t4 {
      then accept; # Do not forget!
```

```
  }
}

[edit interfaces]
lo0 {
  unit 0 {
    family inet {
      filter {
        output loopback-filter;
      }
    }
  }
}
```

NOTE: This is not a complete router configuration. You still have to assign resources to the queues, configure the routing protocols, addresses, and so on.

Configuring Up to 16 Forwarding Classes

By default on all routers, four output queues are mapped to four forwarding classes, as shown in the topic "Default Forwarding Classes" on page 5. On Juniper Networks J Series Services Routers, M120 and M320 Multiservice Edge Routers, and T Series Core Routers, you can configure more than four forwarding classes and queues. For information about configuring J Series routers, see the J Series router documentation.

NOTE: You cannot use CoS-based forwarding features if you configure more than eight forwarding classes on the device.

On M120, M320, MX Series, T Series routers, and PTX Series Packet Transport Routers, you can configure up to 16 forwarding classes and eight queues, with multiple forwarding classes assigned to single queues. The concept of assigning multiple forwarding classes to a queue is sometimes referred to as creating *forwarding-class aliases*. This section explains how to configure M320 and T Series routers.

Mapping multiple forwarding classes to single queues is useful. Suppose, for example, that forwarding classes are set based on multifield packet classification, and the multifield classifiers are different for core-facing interfaces and customer-facing interfaces. Suppose you need four queues for a core-facing interface and five queues for a customer-facing interface, where **fc0** through **fc4** correspond to the classifiers for the customer-facing interface, and **fc5** through **fc8** correspond to classifiers for the core-facing interface, as shown in Figure 1 on page 25.

Figure 1: Customer-Facing and Core-Facing Forwarding Classes

In this example, there are nine classifiers and, therefore, nine forwarding classes. The forwarding class-to-queue mapping is shown in Table 5 on page 25.

Table 5: Sample Forwarding Class-to-Queue Mapping

Forwarding Class Names	Queue Number
fc0 **fc5**	0
fc1 **fc6**	1
fc2 **fc7**	2
fc3 **fc8**	3
fc4	4

To configure up to 16 forwarding classes, include the **class** and **queue-num** statements at the **[edit class-of-service forwarding-classes]** hierarchy level:

```
[edit class-of-service forwarding-classes]
class class-name queue-num queue-number;
```

You can configure up 16 different forwarding-class names. The corresponding output queue number can be from 0 through 7. Therefore, you can map multiple forwarding classes to a single queue. If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler (at the **[edit class-of-service scheduler-maps *map-name* forwarding-class *class-name* scheduler *scheduler-name*]** hierarchy level).

When you configure up to 16 forwarding classes, you can use them as you can any other forwarding class—in classifiers, schedulers, firewall filters (multifield classifiers), policers, and rewrite rules.

When you configure up to 16 forwarding classes, the following limitations apply:

- The **class** and **queue** statements at the **[edit class-of-service forwarding-classes]** hierarchy level are mutually exclusive. In other words, you can include one or the other of the following configurations, but not both:

```
[edit class-of-service forwarding-classes]
queue queue-number class-name;

[edit class-of-service forwarding-classes]
class class-name queue-num queue-number;
```

- On T Series routers only, when you configure IEEE 802.1p rewrite marking on Gigabit Ethernet IQ, Gigabit Ethernet IQ2, Gigabit Ethernet Enhanced IQ (IQE), and Gigabit Ethernet Enhanced IQ2 (IQ2E) PICs, you cannot configure more than eight forwarding classes. This limitation does not apply to M Series routers. On M Series routers, you can configure up to 16 forwarding classes when you configure IEEE 802.1p rewrite marking on any of these PICs.
- For GRE and IP-IP tunnels, IP precedence and DSCP rewrite marking of the inner header do not work with more than eight forwarding classes.
- When you use CoS-based forwarding features, you cannot configure more than eight forwarding classes with a forwarding policy. However, if you try to configure CoS-based forwarding with more than eight forwarding classes configured, commit fails with a message. Therefore, you can configure CBF on a router with eight or less than eight forwarding classes only. Under this condition, the forwarding class to queue mapping can be either one-to-one or one-to-many.
- A scheduler map that maps eight different forwarding classes to eight different schedulers can only be applied to interfaces that support eight queues. If you apply this type of scheduler map to an interface that only supports four queues, then the commit will fail.
- We recommend that you configure the statements changing PICs to support eight queues and then applying an eight queue scheduler map in two separate steps. Otherwise, the commit might succeed but the PIC might not have eight queues when the scheduler map is applied, generating an error.

You can determine the ID number assigned to a forwarding class by issuing the **show class-of-service forwarding-class** command. You can determine whether the classification is fixed by issuing the **show class-of-service forwarding-table classifier mapping** command. In the command output, if the **Table Type** field appears as **Fixed**, the classification is fixed. For more information about fixed classification, see "Applying Forwarding Classes to Interfaces" on page 20.

For information about configuring eight forwarding classes on ATM2 IQ interfaces, see *Enabling Eight Queues on ATM Interfaces*.

This section discusses the following topics:

Enabling Eight Queues on Interfaces

By default, Intelligent Queuing (IQ), Intelligent Queuing 2 (IQ2), Intelligent Queuing Enhanced (IQE), and Intelligent Queuing 2 Enhanced (IQ2E) PICs on M320 and T Series routers are restricted to a maximum of four egress queues per interface. To configure a maximum of eight egress queues on these interfaces, include the **max-queues-per-interface** statement at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

```
[edit chassis fpc slot-number pic pic-number]
max-queues-per-interface (4 | 8);
```

On a TX Matrix or TX Matrix Plus router, include the **max-queues-per-interface** statement at the **[edit chassis lcc *number* fpc *slot-number* pic *pic-number*]** hierarchy level:

```
[edit chassis lcc number fpc slot-number pic pic-number]
max-queues-per-interface (4 | 8);
```

The numerical value can be **4** or **8**.

For Juniper Networks J Series routers, this statement is not supported. J Series routers always have eight queues available.

NOTE: **In addition to configuring eight queues at the [edit chassis] hierarchy level, the configuration at the [edit class-of-service] hierarchy level must support eight queues per interface.**

The maximum number of queues per IQ PIC can be **4** or **8**. If you include the **max-queues-per-interface** statement, all ports on the IQ PIC use configured mode and all interfaces on the IQ PIC have the same maximum number of queues.

To determine how many queues an interface supports, you can check the **CoS queues** output field of the **show interfaces *interface-name* extensive** command:

```
user@host> show interfaces so-1/0/0 extensive
CoS queues: 8 supported
```

If you include the **max-queues-per-interface 4** statement, you can configure all four ports and configure up to four queues per port.

For 4-port OC3c/STM1 Type I and Type II PICs on M320 and T Series routers, when you include the **max-queues-per-interface 8** statement, you can configure up to eight queues on ports 0 and 2. After you commit the configuration, the PIC goes offline and comes back online with only ports 0 and 2 operational. No interfaces can be configured on ports 1 and 3.

For Quad T3 and Quad E3 PICs, when you include the **max-queues-per-interface 8** statement, you can configure up to eight queues on ports 0 and 2. After you commit the configuration, the PIC goes offline and comes back online with only ports 0 and 2 operational. No interfaces can be configured on ports 1 and 3.

When you include the **max-queues-per-interface** statement and commit the configuration, all physical interfaces on the IQ PIC are deleted and readded. Also, the PIC is taken offline and then brought back online immediately. You do not need to take the PIC offline and online manually. You should change modes between four queues and eight queues only when there is no active traffic going to the IQ PIC.

Multiple Forwarding Classes and Default Forwarding Classes

For queues 0 through 3, if you assign multiple forwarding classes to a single queue, default forwarding class assignment works as follows:

- The first forwarding class that you assign to queue 0 acquires the default BE classification and scheduling.
- The first forwarding class that you assign to queue 1 acquires the default EF classification and scheduling.
- The first forwarding class that you assign to queue 2 acquires the default AF classification and scheduling.
- The first forwarding class that you assign to queue 3 acquires the default NC classification and scheduling.

Of course you can override the default classification and scheduling by configuring custom classifiers and schedulers.

If you do not explicitly map forwarding classes to queues 0 through 3, then the respective default classes are automatically assigned to those queues. When you are counting the 16 forwarding classes, you must include in the total any default forwarding classes automatically assigned to queues 0 through 3. As a result, you can map up to 13 forwarding classes to a single queue when the single queue is queue 0, 1, 2, or 3. You can map up to 12 forwarding classes to a single queue when the single queue is queue 4, 5, 6, or 7. In summary, there must be at least one forwarding class each (default or otherwise) assigned to queue 0 through 3, and you can assign the remaining 12 forwarding classes (16–4) to any queue.

For example, suppose you assign two forwarding classes to queue 0 and you assign no forwarding classes to queues 1 through 3. The software automatically assigns one default forwarding class each to queues 1 through 3. This means 11 forwarding classes (16–5) are available for you to assign to queues 4 through 7.

For more information about forwarding class defaults, see "Default Forwarding Classes" on page 5.

PICs Restricted to Four Queues

Some Juniper Networks T Series Core Router PICs support up to 16 forwarding classes and are restricted to 4 queues. Contact Juniper Networks customer support for a current list of T Series router PICs that are restricted to four queues. To determine how many queues an interface supports, you can check the **CoS queues** output field of the **show interfaces *interface-name* extensive** command:

```
user@host> show interfaces so-1/0/0 extensive
CoS queues: 8 supported
```

By default, for T Series router PICs that are restricted to four queues, the router overrides the global configuration based on the following formula:

Qr = Qd mod Rmax

Qr is the queue number assigned if the PIC is restricted to four queues.

Qd is the queue number that would have been mapped if this PIC were not restricted.

Rmax is the maximum number of restricted queues available. Currently, this is four.

For example, assume you map the forwarding class **ef** to queue 6. For a PIC restricted to four queues, the queue number for forwarding class **ef** is **Qr** = 6 mod 4 = 2.

To determine which queue is assigned to a forwarding class, use the **show class-of-service forwarding-class** command from the top level of the CLI. The output shows queue assignments for both global queue mappings and restricted queue mappings:

```
user@host> show class-of-service forwarding-class
Forwarding class           Queue     Restricted Queue  Fabric priority
  be                           0           2              low
  ef                           1           2              low
  assured-forwarding           2           2              low
  network-control              3           3              low
```

For T Series router PICs restricted to four queues, you can override the formula-derived queue assignment by including the **restricted-queues** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
restricted-queues {
  forwarding-class class-name queue queue-number;
}
```

You can configure up to 16 forwarding classes. The output queue number can be from 0 through 3. Therefore, for PICs restricted to four queues, you can map multiple forwarding classes to single queues. If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler. This requirement applies to all PICs. The class name you configure at the **[edit class-of-service restricted-queues]** hierarchy level must be either a default forwarding class name or a forwarding class you configure at the **[edit class-of-service forwarding-classes]** hierarchy level.

Examples: Configuring Up to 16 Forwarding Classes

Configure 16 forwarding classes:

Configuring 16 Forwarding Classes

```
[edit class-of-service]
forwarding-classes {
  class fc0 queue-num 0;
  class fc1 queue-num 0;
  class fc2 queue-num 1;
  class fc3 queue-num 1;
  class fc4 queue-num 2;
  class fc5 queue-num 2;
  class fc6 queue-num 3;
  class fc7 queue-num 3;
```

```
    class fc8 queue-num 4;
    class fc9 queue-num 4;
    class fc10 queue-num 5;
    class fc11 queue-num 5;
    class fc12 queue-num 6;
    class fc13 queue-num 6;
    class fc14 queue-num 7;
    class fc15 queue-num 7;
}
```

For PICs restricted to four queues, map four forwarding classes to each queue:

Restricted Queues: Mapping Two Forwarding Classes to Each Queue

```
[edit class-of-service]
restricted-queues {
    forwarding-class fc0 queue 0;
    forwarding-class fc1 queue 0;
    forwarding-class fc2 queue 0;
    forwarding-class fc3 queue 0;
    forwarding-class fc4 queue 1;
    forwarding-class fc5 queue 1;
    forwarding-class fc6 queue 1;
    forwarding-class fc7 queue 1;
    forwarding-class fc8 queue 2;
    forwarding-class fc9 queue 2;
    forwarding-class fc10 queue 2;
    forwarding-class fc11 queue 2;
    forwarding-class fc12 queue 3;
    forwarding-class fc13 queue 3;
    forwarding-class fc14 queue 3;
    forwarding-class fc15 queue 3;
}
```

If you map multiple forwarding classes to a queue, the multiple forwarding classes must refer to the same scheduler:

Configuring a Scheduler Map Applicable to an Interface Restricted to Four Queues

```
[edit class-of-service]
scheduler-maps {
    interface-restricted {
        forwarding-class be scheduler Q0;
        forwarding-class ef scheduler Q1;
        forwarding-class ef1 scheduler Q1;
        forwarding-class ef2 scheduler Q1;
        forwarding-class af1 scheduler Q2;
        forwarding-class af scheduler Q2;
        forwarding-class nc scheduler Q3;
        forwarding-class nc1 scheduler Q3;
    }
}
[edit class-of-service]
restricted-queues {
    forwarding-class be queue 0;
    forwarding-class ef queue 1;
    forwarding-class ef1 queue 1;
    forwarding-class ef2 queue 1;
    forwarding-class af queue 2;
    forwarding-class af1 queue 2;
    forwarding-class nc queue 3;
```



```
    forwarding-class nc1 queue 3;
}
```

Example: Configuring Different Queuing and Marking Defaults for Outbound Routing Engine and Distributed Protocol Handler Traffic

This example shows how to configure a supported router in an IPv4 network so that traffic generated by the Routing Engine and traffic generated by the distributed protocol handler are assigned to different non-default queues and marked with different nondefault DiffServ code point (DSCP) bits on all egress interfaces.

This configuration enables you to configure network-wide prioritization to control plane protocol hello packets and keepalive packets generated by the router. This feature is supported for egress interfaces hosted on MX Series routers, M120 routers, and Enhanced III FPCs in M320 routers.

Requirements

This example uses the following hardware and software components:

- Two MX80 routers, R1 and R2, each with a 20-port Gigabit Ethernet MIC with SFP. The two routers are directly connected over an IPv4 network.
- Junos OS Release 13.2 or later.

Before you configure this example, configure a Bidirectional Forwarding Detection (BFD) session from port ge-1/0/19 on Router R1 and port ge-1/1/0 on Router R2.

Overview

In this example, you configure an MX80 router in an IPv4 network so that traffic generated by the Routing Engine and traffic generated by the distributed protocol handler are assigned to different nondefault queues and marked with different nondefault DSCP bits.

- Distributed protocol handler sourced traffic is placed in queue 7 on all egress interfaces. Of those packets, Layer 3 packets are marked at egress with DSCP bits 001010.
- Routing Engine sourced traffic is placed in queue 6 on all egress interfaces. Of those packets, Layer 3 packets are marked at egress with DSCP bits 000011.

Because the MX80 router in this example has interfaces hosted on a 20-port Gigabit Ethernet MIC with SFP, you can override the default queuing and DSCP marking behavior of host outbound traffic by including configuration statements at the **[edit class-of-service host-outbound-traffic]** hierarchy level. In this example, you use the **forwarding-class** and

dscp-code-point statements to specify the override values for traffic generated by the distributed protocol handler.

NOTE: This configuration also affects traffic generated by the Routing Engine.

To configure different queuing and DSCP marking of Routing Engine sourced traffic, you must apply a second override configuration. You configure an IPv4 firewall filter that uses the **forwarding-class** and **dscp** actions to specify the override values, and you attach that filter to the egress of the router loopback address. This configuration affects the Routing Engine sourced traffic but not the distributed protocol handler sourced traffic.

Configuration

To configure different queuing and DSCP marking defaults for egress Routing Engine and distributed protocol handler traffic, perform these tasks:

- Configuring R1 Packet Counting on page 32
- Configuring R2 Queuing and Re-Marking of Host Outbound Traffic on page 33
- Configuring R2 Queuing and Re-Marking of Routing Engine Sourced Traffic on page 33

CLI Quick Configuration

To quickly configure this example, copy the following commands, paste them into a text file, remove any line breaks, change any details necessary to match your network configuration, and then copy and paste the commands into the CLI at the **[edit]** hierarchy level.

Router R1

```
set firewall family inet filter f_bfd_source term 1 from forwarding-class control-traffic then
    count c_sent_bfd
set firewall family inet filter f_bfd_source term 1 then accept
set firewall family inet filter f_bfd_source term 2 from forwarding-class-except control-traffic
    then count c_sent_other
set firewall family inet filter f_bfd_source term 2 then accept
set forwarding-options family inet filter output bfd_source
```

Router R2

```
set class-of-service forwarding-classes queue-num 7 bfd_keepalive
set class-of-service host-outbound-traffic forwarding-class bfd_keepalive
set class-of-service host-outbound-traffic dscp-code-point 110000
set class-of-service forwarding-classes queue-num 6 re_control
set firewall family inet filter f_out_loopback term 1 then forwarding-class re_control
set firewall family inet filter f_out_loopback term 1 then dscp 001010
set firewall family inet filter f_out_loopback term 1 then accept
set interfaces lo0 unit 0 family inet filter output f_out_loopback
```

Configuring R1 Packet Counting

Step-by-Step Procedure

To configure Router R1 to count packets that arrive marked for the **network-control** forwarding class:

1. Configure the IPv4 firewall filter term that counts packets marked for the **network-control** forwarding class.

   [edit]

```
user@R1# set firewall family inet filter f_bfd_source term 1 from forwarding-class
  control-traffic then count c_sent_bfd
user@R1# set firewall family inet filter f_bfd_source term 1 then accept
```

2. Configure the IPv4 firewall filter term that counts all other packets.

```
[edit]
user@R1# set firewall family inet filter f_bfd_source term 2 from
  forwarding-class-except control-traffic then count c_sent_other
user@R1# set firewall family inet filter f_bfd_source term 2 then accept
```

3. Apply the firewall filter to all egress packets.

```
[edit]
user@R1# set forwarding-options family inet filter output bfd_source
```

Configuring R2 Queuing and Re-Marking of Host Outbound Traffic

Step-by-Step Procedure

To configure Router R2 to place host outbound traffic in queue 7 and re-mark Layer 3 packets with DSCP bits 110000:

1. Define the **bfd_keepalive** forwarding class and map it to queue 7.

```
[edit]
user@R2# set class-of-service forwarding-classes queue-num 7 bfd_keepalive
```

2. Configure the router to place distributed protocol handler sourced traffic (and also Routing Engine sourced traffic) in queue 7 on all egress interfaces.

```
[edit]
user@R2# set class-of-service host-outbound-traffic forwarding-class bfd_keepalive
```

3. Configure the router to re-mark Layer 3 distributed protocol handler sourced traffic (and also Routing Engine sourced traffic) with DSCP bits 110000, which is compatible with ToS bits 1100 0000.

```
[edit]
user@R2# set class-of-service host-outbound-traffic dscp-code-point 110000
```

Configuring R2 Queuing and Re-Marking of Routing Engine Sourced Traffic

Step-by-Step Procedure

To configure Router R2 to place Routing Engine sourced traffic only in queue 6 and re-mark Layer 3 packets with DSCP bits 001010:

1. Define the **re_control** forwarding class and map it to queue 6.

```
[edit]
user@R2# set class-of-service forwarding-classes queue-num 6 re_control
```

2. Define the IPv4 firewall filter **f_out_loopback** that places matched packets in queue 6, re-marks matched Layer 3 packets with DSCP bits 001010, and accepts all matched packets.

```
[edit]
user@R2# set firewall family inet filter f_out_loopback term 1 then forwarding-class
  re_control
user@R2# set firewall family inet filter f_out_loopback term 1 then dscp 001010
user@R2# set firewall family inet filter f_out_loopback term 1 then accept
```

3. Attach the filter to the output of the router's loopback address so that the filter actions apply to Routing Engine sourced traffic only.

```
[edit]
user@R2# set interfaces lo0 unit 0 family inet filter output f_out_loopback
```

4. If you are done configuring the device, commit the configuration.

```
[edit]
user@R2# commit
```

Results From configuration mode, confirm your configuration by entering the **show class-of-service**, **show firewall**, **show forwarding-options**, and **show interfaces** commands. If the output does not display the intended configuration, repeat the configuration instructions in this example to correct it.

Router R1

```
user@R1# show firewall
family inet {
  filter f_bfd_source {
    term 1 {
      from {
        forwarding-class control-traffic;
      }
      then {
        count c_sent_bfd;
        accept;
      }
    }
    term 2 {
      from {
        forwarding-class-except control-traffic;
      }
      then {
        count c_sent_other;
        accept;
      }
    }
  }
}

user@R1# show forwarding-options
family inet {
  filter {
    output bfd_source;
  }
}
```

Router R2

```
user@R2# show class-of-service
forwarding-classes {
  queue-num 6 re_control;
  queue-num 7 bfd_keepalive;
}
host-outbound-traffic {
  forwarding-class bfd_keepalive;
  dscp-code-point 110000;
}
```



```
user@R2# show firewall
family inet {
   filter f_out_loopback {
      term 1 {
         then {
            forwarding-class re_control;
            dscp 001010;
            accept;
         }
      }
   }
}
user@R2# show interfaces
lo0 {
   unit 0 {
      family inet {
         filter {
            output f_out_loopback;
         }
      }
   }
}
```

Verification

Before you begin verification, enable BFD sessions on both routers.

Confirm that the configuration is working properly.

- Verifying the Queue Assignment of the Traffic That R1 Is Sending in the BFD Session on page 35
- Verifying That Router R1 Is Sending BFD Traffic on page 36
- Verifying That Router R2 Is Receiving BFD Traffic on page 37

Verifying the Queue Assignment of the Traffic That R1 Is Sending in the BFD Session

Purpose Verify the class of service (CoS) forwarding class assignments and type of traffic sent from the BFD source endpoint on Router R1.

Action From operational mode on Router R1, check that BFD packets are sent out the session endpoint on Router R1. With no CoS configuration present, the command output displays statistics about queued and transmitted traffic for the four forwarding classes and four egress queues in use.

```
user@R1> show interfaces queue ge-1/0/19 egress
Physical interface: ge-1/0/19, Enabled, Physical link is Up
  Interface index: 175, SNMP ifIndex: 121
Forwarding classes: 8 supported, 4 in use
Egress queues: 4 supported, 4 in use
Queue: 0, Forwarding classes: best-effort
  Queued:
    ...
  Transmitted:
    ...
Queue: 1, Forwarding classes: expedited-forwarding
  Queued:
    ...
 Transmitted:
    ...
Queue: 2, Forwarding classes: assured-forwarding
  Queued:
    ...
  Transmitted:
    ...
Queue: 3, Forwarding classes: network-control
  Queued:
    ...
  Transmitted:
    ...
```

Meaning The statistics for egress queue 3 reflect BFD session traffic sent to Router R2.

Verifying That Router R1 Is Sending BFD Traffic

Purpose Verify that Router R1 is sending BFD packets from its BFD session endpoint.

Action From operational mode on Router R1, check that the count of BFD packets that R1 sends out the BFD session endpoint continues to increment.

```
user@R1> clear firewall filter f_bfd_source
user@R1> show firewall filter f_bfd_source
Filter: bfd_source
Counters:
Name                                Bytes              Packets
c_sent_bfd                           2770                   70
c_sent_other                            0                    0

user@R1> show firewall filter f_bfd_source
Filter: bfd_source
Counters:
Name                                Bytes              Packets
c_sent_bfd                        2182022                39482
c_sent_other                            0                    0
```

Verifying That Router R2 Is Receiving BFD Traffic

Purpose Verify that Router R2 is receiving BFD packets at its BFD session endpoint.

Action From operational mode on router R2, check that the BFD session endpoint receives packets destined for the Routing Engine with DSCP bits set to 110000, the default DSCP CoS value for the **network-control** forwarding class. The DSCP bits 110000 map to ToS bits 1100 0000, or 0xC0.

```
user@R2> monitor traffic extensive ge-1/1/0 layer2-headers
Address resolution is ON. Use <no-resolve> to avoid any reverse lookup delay.
Address resolution timeout is 4s.
Listening on ge-1/1/0, capture size 1514 bytes

03:23:10.830472 bpf_flags 0x83,  In
        Juniper PCAP Flags [Ext, no-L2, In], PCAP Extension(s) total length 16
          Device Media Type Extension TLV #3, length 1, value: Ethernet (1)
          Logical Interface Encapsulation Extension TLV #6, length 1, value:
Ethernet (14)
          Device Interface Index Extension TLV #1, length 2, value: 132
          Logical Interface Index Extension TLV #4, length 4, value: 68
        -----original packet-----
        PFE proto 2 (ipv4): (tos 0xc0, ttl 255, id 1511, offset 0, flags [none],
 proto: UDP (17), length: 52) 100.1.1.1.bfd-src > 100.1.1.2.bfd-ip: [udp sum ok]

        BFDv1, length: 24
       One-hop Control, State Up, Flags: [Control Plane Independent], Diagnostic:
 No Diagnostic (0x00)
        Detection Timer Multiplier: 3 (30000 ms Detection time), BFD Length: 24
        My Discriminator: 0x00000002, Your Discriminator: 0x00000001
          Desired min Tx Interval:    10000 ms
          Required min Rx Interval:   10000 ms
          Required min Echo Interval:    0 ms
```

Meaning The example input packet entry confirms that the original packet was marked with **tos 0xC0**, which correlates to the default forwarding class **network-control**.

Related Documentation

- Understanding Queuing and Marking of Host Outbound Traffic on page 9
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13
- *monitor traffic*
- *show firewall*
- *show interfaces queue*

CHAPTER 5

CoS-Based Forwarding Configuration

Configuring CoS-Based Forwarding

You can apply CoS-based forwarding (CBF) only to a defined set of routes. Therefore you must configure a policy statement as in the following example:

```
[edit policy-options]
policy-statement my-cos-forwarding {
   from {
      route-filter destination-prefix match-type;
   }
   then {
      cos-next-hop-map map-name;
   }
}
```

This configuration specifies that routes matching the route filter are subject to the CoS next-hop mapping specified by ***map-name***. For more information about configuring policy statements, see the *Routing Policy Feature Guide for Routing Devices*.

NOTE: On M Series routers (except the M120 and M320 routers), forwarding-class-based matching and CBF do not work as expected if the forwarding class has been set with a multifield filter on an input interface.

You can configure CBF on a routing device with eight or less than eight forwarding classes only. Under this condition, the forwarding class to queue mapping can be either one-to-one or one-to-many. However, you cannot configure CBF when the number of forwarding classes configured exceeds eight. Similarly, with CBF configured, you cannot configure more than eight forwarding classes.

To specify a CoS next-hop map, include the **forwarding-policy** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
forwarding-policy {
  next-hop-map map-name {
    forwarding-class class-name {
      next-hop [ next-hop-name ];
      lsp-next-hop [ lsp-regular-expression ];
      discard;
    }
  }
}
```

When you configure CBF with OSPF as the interior gateway protocol (IGP), you must specify the next hop as an interface name or next-hop alias, not as an IP address. This is true because OSPF adds routes with the interface as the next hop for point-to-point interfaces; the next hop does not contain the IP address. For an example configuration, see "Example: Configuring CoS-Based Forwarding" on page 42.

For Layer 3 VPNs, when you use class-based forwarding for the routes received from the far-end provider-edge (PE) router within a VRF instance, the software can match the routes based on the attributes that come with the received route only. In other words, the matching can be based on the route within RIB-in. In this case, the **route-filter** statement you include at the **[edit policy-options policy-statement my-cos-forwarding from]** hierarchy level has no effect because the policy checks the **bgp.l3vpn.0** table, not the ***vrf*.inet.0** table.

The Junos OS applies the CoS next-hop map to the set of next hops previously defined; the next hops themselves can be located across any outgoing interfaces on the routing device. For example, the following configuration associates a set of forwarding classes and next-hop identifiers:

```
[edit class-of-service forwarding-policy]
next-hop-map map1 {
  forwarding-class expedited-forwarding {
    next-hop next-hop1;
    next-hop next-hop2;
  }
  forwarding-class best-effort {
    next-hop next-hop3;
    lsp-next-hop lsp-next-hop4;
  }
}
```

In this example, **next-hop *N*** is either an IP address or an egress interface for some next hop, and **lsp-next-hop4** is a regular expression corresponding to any next hop with that label. Q1 through Q*N* are a set of forwarding classes that map to the specific next hop. That is, when a packet is switched with Q1 through Q*N*, it is forwarded out the interface associated with the associated next hop.

This configuration has the following implications:

- A single forwarding class can map to multiple standard next hops or LSP next hops. This implies that load sharing is done across standard next hops or LSP next hops servicing the same class value. To make this work properly, the Junos OS creates a list of the equal-cost next hops and forwards packets according to standard load-sharing rules for that forwarding class.
- If a forwarding class configuration includes LSP next hops and standard next hops, the LSP next hops are preferred over the standard next hops. In the preceding example, if both **next-hop3** and **lsp-next-hop4** are valid next hops for a route to which **map1** is applied, the forwarding table includes entry **lsp-next-hop4** only.
- If **next-hop-map** does not specify all possible forwarding classes, the default forwarding class is selected as the default. If the default forwarding class is not specified in the next-hop map, a default is designated randomly. The default forwarding class is the class associated with queue 0.
- For LSP next hops, the Junos OS uses UNIX **regex(3)**-style regular expressions. For example, if the following labels exist: **lsp**, **lsp1**, **lsp2**, **lsp3**, the statement **lsp-next-hop lsp** matches **lsp**, **lsp1**, **lsp2**, and **lsp3**. If you do not desire this behavior, you must use the anchor characters **lsp-next-hop " ^lsp$"**, which match **lsp** only.
- The route filter does not work because the policy checks against the **bgp.l3vpn.0** table instead of the ***vrf*.inet.0** table.

The final step is to apply the route filter to routes exported to the forwarding engine. This is shown in the following example:

```
routing-options {
   forwarding-table {
      export my-cos-forwarding;
   }
}
```

This configuration instructs the routing process to insert routes to the forwarding engine matching **my-cos-forwarding** with the associated next-hop CBF rules.

The following algorithm is used when you apply a configuration to a route:

- If the route is a single next-hop route, all traffic goes to that route; that is, no CBF takes effect.
- For each next hop, associate the proper forwarding class. If a next hop appears in the route but not in the **cos-next-hop** map, it does not appear in the forwarding table entry.
- The default forwarding class is used if all forwarding classes are not specified in the next-hop map. If the default is not specified, one is chosen randomly.

Overriding the Input Classification

For IPv4 or IPv6 packets, you can override the incoming classification, assigning them to the same forwarding class based on their input interface, input precedence bits, or destination address. You do so by defining a policy class when configuring CoS properties and referencing this class when configuring a routing policy.

When you override the classification of incoming packets, any mappings you configured for associated precedence bits or incoming interfaces to output transmission queues are ignored. Also, if the packet loss priority (PLP) bit was set in the packet by the incoming interface, the PLP bit is cleared.

To override the input packet classification, do the following:

1. Define the policy class by including the **class** statement at the **[edit class-of-service policy]** hierarchy level:

   ```
   [edit class-of-service]
   forwarding-policy {
     class class-name {
       classification-override {
         forwarding-class class-name;
       }
     }
   }
   ```

   class-name is a name that identifies the class.

2. Associate the policy class with a routing policy by including it in a **policy-statement** statement at the **[edit policy-options]** hierarchy level. Specify the destination prefixes in the **route-filter** statement and the CoS policy class name in the **then** statement.

   ```
   [edit policy-options]
   policy-statement policy-name {
     term term-name {
       from {
         route-filter destination-prefix match-type <class class-name>
       }
       then class class-name;
     }
   }
   ```

3. Apply the policy by including the **export** statement at the **[edit routing-options]** hierarchy level:

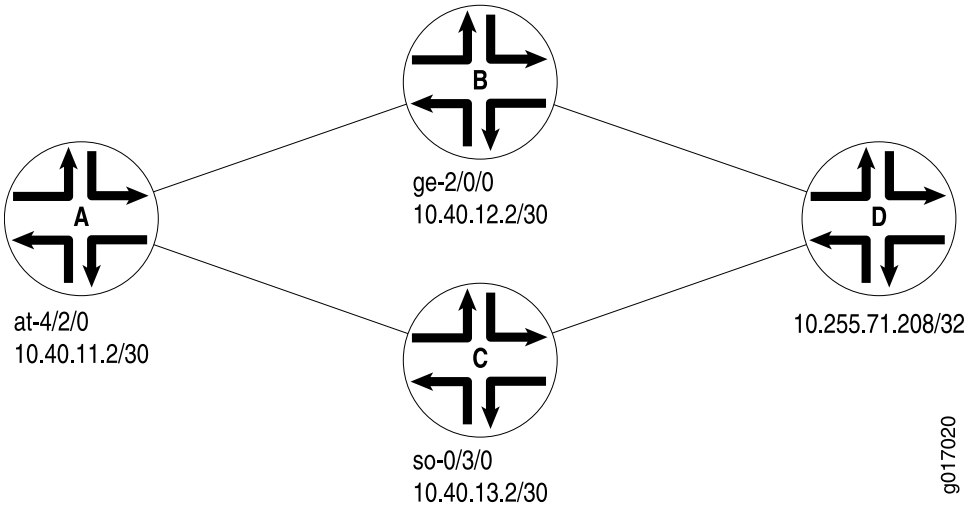
   ```
   [edit routing-options]
   forwarding-table {
     export policy-name;
   }
   ```

Example: Configuring CoS-Based Forwarding

Router A has two routes to destination **10.255.71.208** on Router D. One route goes through Router B, and the other goes through Router C, as shown in Figure 2 on page 43.

Configure Router A with CBF to select Router B for queue 0 and queue 2, and Router C for queue 1 and queue 3.

Figure 2: Sample CoS-Based Forwarding



When you configure CBF with OSPF as the IGP, you must specify the next hop as an interface name, not as an IP address. The next hops in this example are specified as **ge-2/0/0.0** and **so-0/3/0.0**.

```
[edit class-of-service]
forwarding-policy {
  next-hop-map my_cbf {
    forwarding-class be {
      next-hop ge-2/0/0.0;
    }
    forwarding-class ef {
      next-hop so-0/3/0.0;
    }
    forwarding-class af {
      next-hop ge-2/0/0.0;
    }
    forwarding-class nc {
      next-hop so-0/3/0.0;
    }
  }
}
classifiers {
  inet-precedence inet {
    forwarding-class be {
      loss-priority low code-points [ 000 100 ];
    }
    forwarding-class ef {
      loss-priority low code-points [ 001 101 ];
    }
    forwarding-class af {
      loss-priority low code-points [ 010 110 ];
    }
    forwarding-class nc {
      loss-priority low code-points [ 011 111 ];
    }
  }
```

```
}
forwarding-classes {
   queue 0 be;
   queue 1 ef;
   queue 2 af;
   queue 3 nc;
}
interfaces {
   at-4/2/0 {
      unit 0 {
         classifiers {
            inet-precedence inet;
         }
      }
   }
}

[edit policy-options]
policy-statement cbf {
   from {
      route-filter 10.255.71.208/32 exact;
   }
   then cos-next-hop-map my_cbf;
}

[edit routing-options]
graceful-restart;
forwarding-table {
   export cbf;
}

[edit interfaces]
traceoptions {
   file trace-intf size 5m world-readable;
   flag all;
}
so-0/3/0 {
   unit 0 {
      family inet {
         address 10.40.13.1/30;
      }
      family iso;
      family mpls;
   }
}
ge-2/0/0 {
   unit 0 {
      family inet {
         address 10.40.12.1/30;
      }
      family iso;
      family mpls;
   }
}
at-4/2/0 {
   atm-options {
```

```
    vpi 1 {
      maximum-vcs 1200;
    }
  }
  unit 0 {
    vci 1.100;
    family inet {
      address 10.40.11.2/30;
    }
    family iso;
    family mpls;
  }
}
```

Example: Configuring CoS-Based Forwarding for Different Traffic Types

One common use for CoS-based forwarding and next-hop maps is to enforce different handling for different traffic types, such as voice and video. For example, an LSP-based next hop can be used for voice and video, and a non-LSP next-hop can be used for best effort traffic.

Only the forwarding policy is shown in this example:

```
[edit class-of-service]
forwarding-policy {
  next-hop-map ldp-map {
    forwarding-class expedited-forwarding {
      lsp-next-hop voice;
      non-lsp-next-hop;
    }
    forwarding-class assured-forwarding {
      lsp-next-hop video;
      non-lsp-next-hop;
    }
    forwarding-class best-effort {
      non-lsp-next-hop;
      discard;
    }
  }
}
```

Example: Configuring CoS-Based Forwarding for IPv6

This example configures CoS-based forwarding (CBF) next-hop maps and CBF LSP next-hop maps for IPv6 addresses.

You can configure a next-hop map with both IPv4 and IPv6 addresses, or you can configure separate next-hop maps for IPv4 and IPv6 addresses and include the **from family (inet | inet6)** statements at the **[edit policy-options policy-options policy-statement *policy-name* term *term-name*]** hierarchy level to ensure that only next-hop maps of a specified protocol are applied to a specified route.

If you do not configure separate next-hop maps and include the **from family (inet | inet6)** statements in the configuration, when a route uses two next hops (whether IPv4, IPv6,

interface, or LSP next hop) in at least two of the specified forwarding classes, CBF is used for the route; otherwise, the CBF policy is ignored.

1. Define the CBF next-hop map:

```
[edit class-of-service]
forwarding-policy {
  next-hop-map cbf-map {
    forwarding-class best-effort {
      next-hop [ ::192.168.139.38 192.168.139.38 ];
    }
    forwarding-class expedited-forwarding {
      next-hop [ ::192.168.140.5 192.168.140.5 ];
    }
    forwarding-class assured-forwarding {
      next-hop [ ::192.168.145.5 192.168.145.5 ];
    }
    forwarding-class network-control {
      next-hop [ ::192.168.141.2 192.168.141.2 ];
    }
  }
}
```

2. Define the CBF forwarding policy:

```
[edit policy-options]
policy-statement ls {
  then cos-next-hop-map cbf-map;
}
```

3. Export the CBF forwarding policy:

```
[edit routing-options]
forwarding-table {
  export ls;
}
```

CHAPTER 6

Configuration Statements

class (CoS-Based Forwarding)

Syntax

```
class class-name {
  classification-override {
    forwarding-class class-name;
  }
}
```

Hierarchy Level [edit class-of-service forwarding-policy]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure CoS-based forwarding class.

Options *class-name*—Name of the routing policy class.

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

- Overriding the Input Classification on page 41

class (Forwarding Classes)

Syntax class *class-name* queue-num *queue-number* priority (high | low);

Hierarchy Level [edit class-of-service forwarding-classes]

Release Information Statement introduced in Junos OS Release 8.1.

Description On M120 , M320, MX Series routers, T Series routers and EX Series switches only, specify the output transmission queue to which to map all input from an associated forwarding class.

This statement enables you to configure up to 16 forwarding classes with multiple forwarding classes mapped to single queues. If you want to configure up to eight forwarding classes with one-to-one mapping to output queues, use the **queue** statement instead of the **class** statement at the **[edit class-of-service** forwarding-classes**]** hierarchy level.

Options *class-name*—Name of forwarding class.

queue-number—Output queue number.

Range: 0 through 15. Some T Series router PICs are restricted to 0 through 3.

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 Forwarding Classes on page 19
- queue (Global Queues) on page 65

classification-override

Syntax

```
classification-override {
  forwarding-class class-name;
}
```

Hierarchy Level [edit class-of-service forwarding-policy class *class-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For IPv4 packets, override the incoming packet classification, assigning all packets sent to a destination prefix to the same output transmission queue.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Overriding the Input Classification on page 41
- **policy-statement** in the *Junos OS Routing Protocols Library for Routing Devices*

discard (Forwarding Class)

Syntax

```
discard;
```

Hierarchy Level [edit class-of-service forwarding-policy next-hop-map *map-name* forwarding-class *class-name*]

Release Information Statement introduced in Junos OS Release 9.1.

Description Discard traffic sent to this forwarding class for the next-hop map referenced by this forwarding policy.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring CoS-Based Forwarding on page 39
- non-lsp-next-hop on page 62

dscp-code-point (CoS Host Outbound Traffic)

Syntax dscp-code-point *value*;

Hierarchy Level [edit class-of-service host-outbound-traffic]

Release Information
Statement introduced in Junos OS Release 8.4.
Statement introduced in Junos OS 8.5 for SRX1400, SRX3400, SRX3600, SRX5600, and SRX5800 firewalls.
Statement introduced before Junos OS 11.4 for EX Series switches.
Statement introduced before Junos OS 12.1 for SRX1400, SRX3400, SRX3600, SRX5600, and SRX5800 firewalls.
Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.

Description Specify the value of the DSCP bits in the type of service (ToS) field of host outbound traffic (packets generated by the local Routing Engine) as they are placed in the default or specified output queue on all egress interfaces. This statement does not affect transit traffic or incoming traffic.

If you use the **ping** operational mode command with the **tos *type-of-servce*** option, the value specified in this configuration statement overrides the DSCP value you specify in the **ping** command.

NOTE: Any DSCP rewrite rules configured on a 10-Gigabit Ethernet LAN/WAN PIC with SFP+ overwrite this DSCP value.

For egress interfaces hosted on MX Series routers, M120 routers, or Enhanced III FPCs in M320 routers, both Routing Engine sourced traffic and distributed protocol handler traffic are affected. For all other egress interfaces, only Routing Engine sourced traffic is affected.

Options **code-point**—Six-bit DSCP code point value.

Required Privilege Level
interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Overview of BA Classifier Types*
- *Default DSCP and DSCP IPv6 Classifier*
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13.

forwarding-class (Forwarding Policy)

Syntax

```
forwarding-class class-name {
  next-hop [ next-hop-name];
  lsp-next-hop [ lsp-regular-expression ];
  non-lsp-next-hop;
  discard;
}
```

Hierarchy Level [edit class-of-service forwarding-policy next-hop-map *map-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define forwarding class name and associated next hops.

Options ***class-name***—Name of the forwarding class.

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

- Overriding the Input Classification on page 41

forwarding-class (CoS Host Outbound Traffic)

Syntax forwarding-class *class-name*;

Hierarchy Level [edit class-of-service host-outbound-traffic]

Release Information Statement introduced in Junos OS Release 8.4.
Statement introduced in Junos OS 8.5 for SRX1400, SRX3400, SRX3600, SRX5600, and SRX5800 firewalls.
Statement introduced before Junos OS 11.4 for EX Series switches.
Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.

Description Specify the name of the forwarding class to which host outbound traffic is assigned on all egress interfaces. The output queue associated with the forwarding class must be properly configured on all interfaces. In the case of a restricted interface, the traffic flows through a restricted queue.

For egress interfaces hosted on MX Series routers, M120 routers, or Enhanced III FPCs in M320 routers, both Routing Engine sourced traffic and distributed protocol handler traffic are affected. For all other egress interfaces, only Routing Engine sourced traffic is affected.

This statement does not affect transit traffic or incoming traffic.

Default If you do not configure an output queue for host outbound traffic, the router uses the default queue assignments for host outbound traffic.

Options *class-name*—Name of the forwarding class.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Overview of Forwarding Classes on page 3
- Default Queue Assignments for Routing Engine Sourced Traffic on page 11
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13.

forwarding-class (Interfaces)

Syntax forwarding-class *class-name*;

Hierarchy Level [edit class-of-service interfaces *interface-name* unit *logical-unit-number*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.2 for ACX Series routers.

Description Associate a forwarding class configuration or default mapping with a specific interface.

Options ***class-name***—Name of the forwarding class.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Applying Forwarding Classes to Interfaces on page 20

forwarding-class (Restricted Queues)

Syntax forwarding-class *class-name* queue *queue-number*;

Hierarchy Level [edit class-of-service restricted-queues]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320 and T Series routers only, map forwarding classes to restricted queues. You can map up to eight forwarding classes to restricted queues.

Options ***class-name***—Name of the forwarding class.

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.

forwarding-classes (Class-of-Service)

Syntax

```
forwarding-classes {
    class queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low) [ policing-priority (premium | normal) ];
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
policing-priority option introduced in Junos OS Release 9.5.
Statement introduced on PTX Series Packet Transport Routers in Junos OS Release 12.1.

Description Associate the forwarding class with a queue name and number. For M320, MX Series, T Series routers and EX Series switches only, you can configure fabric priority queuing by including the **priority** statement. For Enhanced IQ PICs, you can include the **policing-priority** option.

NOTE: The priority add policing-priority options are not supported on PTX Series Packet Transport Routers.

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 Forwarding Classes on page 19
- Forwarding Classes and Fabric Priority Queues on page 8
- *Example: Configuring CoS for a PBB Network on MX Series Routers*
- *Configuring Layer 2 Policers on IQE PICs*
- Classifying Packets by Egress Interface on page 20

forwarding-class-map

Syntax

```
forwarding-class-map forwarding-class-map-name {
  class class-name queue-num queue-number [ restricted-queue queue-number ];
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced in Junos OS Release 9.6.

Description For the IQ, IQE, LSQ and ATM2 PICs in the T Series routers only and for EX Series switches, configure a forwarding class map for unicast and multicast traffic and a user-configured queue number for an egress interface.

Options

class-name—Name of the forwarding class.

forwarding-class-map-name—Name of the forwarding class map for traffic.

queue-number—Number of the egress queue.

Range: 0 through 3 or 7, depending on chassis and configuration

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Forwarding Classes on page 19
- Classifying Packets by Egress Interface on page 20
- output-forwarding-class-map on page 63

forwarding-policy

Syntax

```
forwarding-policy {
  next-hop-map map-name {
    forwarding-class class-name {
      next-hop [ next-hop-name ];
      lsp-next-hop [ lsp-regular-expression ];
      non-lsp-next-hop;
      discard;
    }
  }
  class class-name {
    classification-override {
      forwarding-class class-name;
    }
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define CoS-based forwarding policy options.

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 CoS-Based Forwarding on page 39

host-outbound-traffic (Class-of-Service)

Syntax

```
host-outbound-traffic {
  forwarding-class class-name;
  dscp-code-point value;
  ieee-802.1 {
    default value;
    rewrite-rules;
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced in Junos OS Release 8.4.
Statement introduced in Junos OS 8.5 for SRX1400, SRX3400, SRX3600, SRX5600, and SRX5800 firewalls.
Statement introduced before Junos OS 11.4 for EX Series switches.
Support for **ieee-802.1** statement introduced in Junos OS Release 12.3.
Support for distributed protocol handler traffic introduced in Junos OS Release 13.2.

Description Classify and mark host outbound traffic. This statement does not affect transit traffic or incoming traffic.

Default If you do not specify a forwarding class or DSCP value, the router uses the default queue and DSCP bit assignments for host outbound traffic.

Options 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

- Default Queue Assignments for Routing Engine Sourced Traffic on page 11
- *Default DSCP and DSCP IPv6 Classifier*
- Changing the Default Queuing and Marking of Host Outbound Traffic on page 13.
- *Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic*
- *Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface*
- *Understanding Junos OS CoS Components for EX Series Switches*

ieee-802.1 (Host Outbound Traffic)

Syntax

```
ieee-802.1 {
  default value;
  rewrite-rules;
}
```

Hierarchy Level [edit class-of-service host-outbound-traffic]

Release Information Statement introduced in Junos OS Release 12.3.

Description Apply the IEEE 802.1p rewrite rules associated with the egress logical interface to the IEEE 802.1p PCP field for all host outbound traffic on that interface.

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Configuring a Global Default IEEE 802.1p Value for All Host Outbound Traffic*
- *Applying Egress Interface Rewrite Rules to the IEEE 802.1p Field for All Host Outbound Traffic on the Interface*
- *Rewriting Packet Header Information Overview*
- *Configuring Rewrite Rules*

interfaces

Syntax

```
interfaces {
  interface-name {
    classifiers{
      dscp(classifier-name | default) {
    }
      ieee-802.1 (classifier-name | default)  vlan-tag (inner | outer | classifier-name);
      inet-precedence (rewrite-name | default);
    }
    input-scheduler-map map-name;
    input-shaping-rate rate;
    irb {
      unit logical-unit-number {
        classifiers {
          type (classifier-name | default);
        }
        rewrite-rules {
          dscp (rewrite-name | default);
          dscp-ipv6 (rewrite-name | default);
          exp (rewrite-name | default)protocol protocol-types;
          ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
          inet-precedence (rewrite-name | default);
        }
      }
    }
    member-link-scheduler (replicate | scale);
    rewrite-rules {
      dscp (rewrite-name | default);
      ieee-802.1 (rewrite-name | default) vlan-tag (outer);
      inet-precedence (rewrite-name | default);
      }
    }
    scheduler-map map-name;
    scheduler-map-chassis map-name;
    shaping-rate rate;
    unit logical-unit-number {
      classifiers {
        type (classifier-name | default) family (mpls | inet);
      }
      forwarding-class class-name;
      fragmentation-map map-name;
      input-shaping-rate (percent percentage | rate);
      input-traffic-control-profile profile-name shared-instance instance-name;
      output-traffic-control-profile profile-name shared-instance instance-name;
      per-session-scheduler;
      rewrite-rules {
        dscp (rewrite-name | default);
        dscp-ipv6 (rewrite-name | default);
        exp (rewrite-name | default) protocol protocol-types;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        inet-precedence (rewrite-name | default);
```

```
        }
        scheduler-map map-name;
        shaping-rate rate;
        translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp
          | to-inet-precedence-from-inet-precedence) table-name;
      }
    }
    interface-set interface-set-name {
      excess-bandwidth-share;
      internal-node;
      output-traffic-control-profile profile-name;
      output-traffic-control-profile-remaining profile-name;
    }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
Interface-set level added in Junos OS Release 8.5.

Description Configure interface-specific CoS properties for incoming packets.

NOTE: The **dscp-ipv6** and **ieee-802.1ad** classifier types are not supported on ACX Series routers. For further information about support on ACX Series routers, see *Understanding CoS CLI Configuration Statements on ACX Series Universal Access Routers*.

Options 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

- *Overview of BA Classifier Types*
- *Configuring Rewrite Rules*
- *Understanding CoS CLI Configuration Statements on ACX Series Universal Access Routers*

lsp-next-hop (CoS-Based Forwarding)

Syntax lsp-next-hop [*lsp-regular-expression*];

Hierarchy Level [edit class-of-service forwarding-policy next-hop-map *map-name* forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the LSP regular expression to which to map forwarded traffic.

Options ***lsp-regular-expression***—Next-hop LSP label.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS-Based Forwarding on page 39

next-hop (Class-Of-Service)

Syntax next-hop [*next-hop-name*];

Hierarchy Level [edit class-of-service forwarding-policy next-hop-map *map-name* forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the next-hop name or address to which to map forwarded traffic.

Options ***next-hop-name***—Next-hop alias or IP address.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS-Based Forwarding on page 39

next-hop-map

Syntax

```
next-hop-map map-name {
  forwarding-class class-name {
    next-hop next-hop-name;
    lsp-next-hop [ lsp-regular-expression ];
    non-lsp-next-hop;
    discard;
  }
}
```

Hierarchy Level [edit class-of-service forwarding-policy]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the map for CoS forwarding routes.

Options *map-name*—Map that defines next-hop routes.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS-Based Forwarding on page 39

non-lsp-next-hop

Syntax

```
non-lsp-next-hop;
```

Hierarchy Level [edit class-of-service forwarding-policy next-hop-map *map-name* forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 9.0.

Description Use a non-LSP next hop for traffic sent to this forwarding class next-hop map of this forwarding policy.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation
- Configuring CoS-Based Forwarding on page 39

output-forwarding-class-map

Syntax output-forwarding-class-map *forwarding-class-map-name*;

Hierarchy Level [edit class-of-service forwarding-class-map]

Release Information Statement introduced in Junos OS Release 9.6.

Description Apply a configured forwarding class map to a logical interface.

Options *forwarding-class-map-name*—Name of a forwarding class mapping configured at the **[edit class-of-service forwarding-class-map]** hierarchy level.

Required Privilege Level interface-control—To add this statement to the configuration.

Related Documentation

- Classifying Packets by Egress Interface on page 20
- forwarding-class-map on page 55

priority (Fabric Priority)

Syntax priority (high | low);

Hierarchy Level [edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*], [edit class-of-service forwarding-classes queue *queue-number class-name*]

Release Information Statement introduced before Junos OS Release 7.4.
[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*] hierarchy level added in Junos OS Release 8.1.

Description For M320 routers, MX Series routers, T Series routers and EX Series switches only, specify a fabric priority value.

The two hierarchy levels are mutually exclusive. To configure up to eight forwarding classes with one-to-one mapping between forwarding classes and output queues, include this statement at the **[edit class-of-service forwarding-classes queue *queue-number class-name*]** hierarchy level. To configure up to 16 forwarding classes with multiple forwarding classes mapped to single queues, include this statement at the **[edit class-of-service forwarding-classes class *class-name* queue-num *queue-number*]** hierarchy level.

Options **low**—Forwarding class's fabric queuing has low priority.

high—Forwarding class's fabric queuing has high priority.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Forwarding Classes and Fabric Priority Queues on page 8
- Configuring Up to 16 Forwarding Classes on page 24

queue (Global Queues)

Syntax queue *queue-number class-name*;

Hierarchy Level [edit class-of-service forwarding-classes]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify the output transmission queue to which to map all input from an associated forwarding class.

On M120, M320, MX Series, T Series routers and on EX Series switches, this statement enables you to configure up to eight forwarding classes with one-to-one mapping to output queues. If you want to configure up to 16 forwarding classes with multiple forwarding classes mapped to single output queues, include the **class** statement instead of the **queue** statement at the **[edit class-of-service forwarding-classes]** hierarchy level.

Options *class-name*—Name of forwarding class.

queue-number—Output queue number.

Range: For M Series routers, 0 through 3. For M120, M320, MX Series, T Series routers and EX Series switches, 0 through 7. Some T Series router PICs are restricted to 0 through 3.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- Configuring Forwarding Classes on page 19
- class (Forwarding Classes) on page 48

queue (Restricted Queues)

Syntax queue *queue-number*;

Hierarchy Level [edit class-of-service restricted-queues forwarding-class *class-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.

Options *queue-number*—Output queue number.

Range: 0 through 3.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

restricted-queues

Syntax

```
restricted-queues {
    forwarding-class class-name queue queue-number;
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.

Description For M320, MX Series, T Series routers and EX Series switches only, map forwarding classes to restricted queues.

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.

unit

Syntax

```
unit logical-unit-number {
  classifiers {
    type (classifier-name | default) family (mpls | all);
  }
  forwarding-class class-name;
  fragmentation-map map-name;
  input-traffic-control-profile profile-name shared-instance instance-name;
  output-traffic-control-profile profile-name shared-instance instance-name;
  per-session-scheduler;
  rewrite-rules {
    dscp (rewrite-name | default);
    dscp-ipv6 (rewrite-name | default);
    exp (rewrite-name | default)protocol protocol-types;
    exp-push-push-push default;
    exp-swap-push-push default;
    ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
    inet-precedence (rewrite-name | default);
  }
  scheduler-map map-name;
  shaping-rate rate;
}
```

Hierarchy Level [edit class-of-service interfaces *interface-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a logical interface on the physical device. You must configure a logical interface to be able to use the physical device.

Options ***logical-unit-number***—Number of the logical unit.

Range: 0 through 16,384

The remaining statements are explained separately.

Required Privilege Level interface—To view this statement in the configuration.
interface-control—To add this statement to the configuration.

Related Documentation

- *Overview of BA Classifier Types*
- *Configuring Rewrite Rules*

PART 3

Index

Index

Symbols

A

B

C

D

E

F