Tricolor Marking Policers on EX9200 Switches

Published: 2013-04-19

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

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:

- EX 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

Tricolor Marking Policers

Traffic Policing Overview

This topic covers the following information:

Congestion Management for IP Traffic Flows

Traffic policing, also known *rate limiting*, is an essential component of network access security that is designed to thwart denial-of-service (DoS) attacks. Traffic policing enables you to control the maximum rate of IP traffic sent or received on an interface and also to partition network traffic into multiple priority levels, also known as *classes of service*. A policer defines a set of traffic rate limits and sets consequences for traffic that does not conform to the configured limits. Packets in a traffic flow that does not conform to traffic limits are either discarded or marked with a different forwarding class or packet loss priority (PLP) level.

With the exception of policers configured to rate-limit aggregate traffic (all protocol families and logical interfaces configured on a physical interface), you can apply a policer to all IP packets in a Layer 2 or Layer 3 traffic flow at a logical interface.

With the exception of policers configured to rate-limit based on physical interface media rate, you can apply a policer to specific IP packets in a Layer 3 traffic flow at a logical interface by using a stateless firewall filter.

You can apply a policer to inbound or outbound interface traffic. Policers applied to inbound traffic help to conserve resources by dropping traffic that does not need to be

routed through a network. Dropping inbound traffic also helps to thwart denial-of-service (DoS) attacks. Policers applied to outbound traffic control the bandwidth used.

NOTE: Traffic policers are instantiated on a per-PIC basis. Traffic policing does not work when the traffic for one local policy decision function (L-PDF) subscriber is distributed over multiple Multiservices PICs in an AMS group.

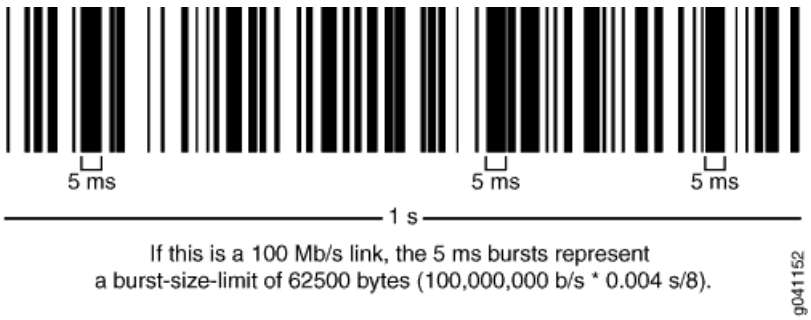
Traffic Limits

Junos® OS policers use the *token-bucket* algorithm to enforce a limit on average transmit or receive rate of IP traffic at an interface while allowing bursts of traffic up to a maximum value based on the overall traffic load. The token-bucket algorithm offers more flexibility than the *leaky-bucket* algorithm in that you can allow a specified amount of bursting before starting to discard packets or apply a penalty to packet output-queuing priority or packet drop priority.

In the token-bucket model, the bucket represents the policing function. Tokens are added to the bucket at a fixed rate, but only up to the specified depth of the bucket. Each token represents a "credit" for some number of bits, and tokens in the bucket are "cashed in" for the ability to transmit or receive traffic at the interface. When sufficient tokens are present in the bucket, a traffic flow continues unrestricted. Otherwise, packets might be dropped or else re-marked with a lower forwarding class, a higher packet loss priority (PLP) level, or both.

- The rate at which tokens are added to the bucket represents the highest average transmit or receive rate in bits per second allowed for a given service level. You specify this highest average traffic rate as the *bandwidth limit* of the policer. If the traffic arrival rate is so high that at some point insufficient tokens are present in the bucket, then the traffic flow is no longer conforming to the traffic limit.
- The depth of the bucket in bytes controls the amount of back-to-back bursting allowed. You specify this factor as the *burst-size limit* of the policer. This second limit affects the average transmit or receive rate by limiting the number of bytes permitted in a transmission burst for a given interval of time. Bursts exceeding the current burst-size limit are dropped until there are sufficient tokens available to permit the burst to proceed.

Figure 1: Network Traffic and Burst Rates



As shown in the figure above, a UPC bar code is a good facsimile of what traffic looks like on the line; an interface is either transmitting (bursting at full rate) or it is not. The black lines represent periods of data transmission and the white space represents periods of silence when the token bucket can replenish.

Depending on the type of policer used, packets in a policed traffic flow that surpasses the defined limits might be implicitly set to a higher PLP level, assigned to a configured forwarding class or set to a configured PLP level (or both), or simply discarded. If packets encounter downstream congestion, packets with a **low** PLP level are less likely to be discarded than those with a **medium-low**, **medium-high**, or **high** PLP level.

Traffic Color Marking

Based on the particular set of traffic limits configured, a policer identifies a traffic flow as belonging to one of either two or three categories that are similar to the colors of a traffic light used to control automobile traffic.

A *two-color-marking* policer categorizes traffic as either conforming to the traffic limits (green) or violating the traffic limits (red):

- Green—Two-color-marking policers implicitly set the packets in a green flow to the low PLP level, and you cannot configure any policer actions for conforming traffic.
- Red—Two-color-marking policers do not perform any implicit actions on packets in a red flow. Instead, those packets are handled according to the actions specified in the policer configuration. You can configure a two-color-marking policer to simply discard packets if the traffic flow is red. Alternatively, you can configure a two-color-marking policer to handle the packets in a red flow by setting the PLP level to either **low** or **high**, assigning the packets to any forwarding class already configured, or both.

  On MX Series, M120, and M320 routers and M7i and M10i routers with the Enhanced CFEB (CFEB-E) and EX Series switches only, you can specify two additional PLP levels for packets in a red flow: **medium-low** or **medium-high**.

Three-color-marking policers categorize traffic as conforming to the traffic limits (green), violating the traffic limits (red), or exceeding the traffic limits but within an allowed range (yellow):

- Green—Like two-color-marking policers, three-color-marking policers implicitly set the packets in a green flow to the low PLP level, and you cannot configure any policer actions for conforming traffic.
- Yellow—Unlike two-color-marking policers, three-color-marking policers categorize a second type of nonconforming traffic: yellow.

  Single-rate three-color policing categorizes as yellow traffic that exceeds the traffic limits while conforming to a second defined burst-size limit. Two-rate three-color policing categorizes as yellow traffic that exceeds the traffic limits while conforming to both a second defined burst-size limit and a second defined bandwidth limit.

  Three-color-marking policers implicitly set the packets in a yellow flow to the medium-high PLP level so that the packets incur a less severe penalty than those in a red flow. You cannot configure any policer actions for yellow traffic.

- Red—Unlike two-color-marking policers, three-color-marking policers implicitly set the packets in a red flow to the high PLP level, which is the highest PLP value. You can also configure a three-color-marking policer to discard the packets in a red flow instead of forwarding them with a high PLP setting.

Two-color-marking policers allows bursts of traffic for short periods, whereas three-color-marking policers allow more sustained bursts of traffic.

Forwarding Classes and PLP Levels

A packet's forwarding class assignment and PLP level are used by the Junos OS class of service (CoS) features. The Junos CoS features include a set of mechanisms that you can use to provide differentiated services when best-effort traffic delivery is insufficient. For router (and switch) interfaces that carry IPv4, IPv6, and MPLS traffic, you can configure CoS features to take in a single flow of traffic entering at the edge of your network and provide different levels of service across the network—internal forwarding and scheduling (queuing) for output—based on the forwarding class assignments and PLP levels of the individual packets.

NOTE: **Forwarding-class or loss-priority assignments performed by a policer or a stateless firewall filter override any such assignments performed on the ingress by the CoS default IP precedence classification at all logical interfaces or by any configured behavior aggregate (BA) classifier that is explicitly mapped to a logical interface.**

Based on CoS configurations, packets of a given forwarding class are transmitted through a specific output queue, and each output queue is associated with a transmission service level defined in a *scheduler*.

Based on other CoS configurations, when packets in an output queue encounter congestion, packets with higher loss-priority values are more likely to be dropped by the random early detection (RED) algorithm. Packet loss priority values affect the scheduling of a packet without affecting the packet's relative ordering within the traffic flow.

Policer Application to Traffic

After you have defined and named a policer, it is stored as a template. You can later use the same policer name to provide the same policer configuration each time you want to use it. This eliminates the need to define the same policer values more than once.

You can apply a policer to a traffic flow in either of two ways:

- You can configure a standard stateless firewall filter that specifies the **policer *policer-name*** nonterminating action or the **three-color-policer (single-rate | two-rate) *policer-name*** nonterminating action. When you apply the standard filter to the input or output at a logical interface, the policer is applied to all packets of the filter-specific protocol family that match the conditions specified in the filter configuration.

With this method of applying a policer, you can define specific classes of traffic on an interface and apply traffic rate-limiting to each class.

- You can apply a policer directly to an interface so that traffic rate-limiting applies to all traffic on that interface, regardless of protocol family or any match conditions.

You can configure policers at the queue, logical interface, or Layer 2 (MAC) level. Only a single policer is applied to a packet at the egress queue, and the search for policers occurs in this order:

- Queue level
- Logical interface level
- Layer 2 (MAC) level

Related Documentation

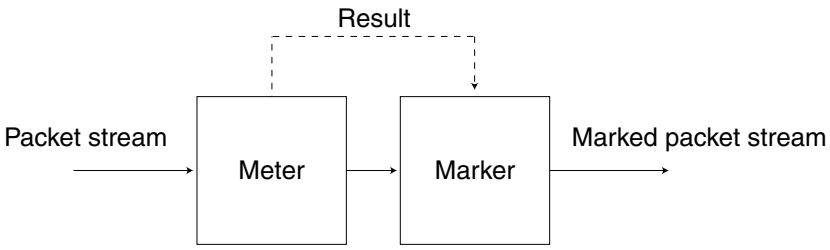
- Stateless Firewall Filter Overview.
- Traffic Policer Types
- Order of Policer and Firewall Filter Operations
- Packet Flow Through the CoS Process Overview

Tricolor Marking Architecture

Policers provide two functions: metering and marking.

The policer meters each packet and passes the packet and the metering result to the marker, as shown in Figure 2 on page 7.

Figure 2: Flow of Tricolor Marking Policer Operation



The meter operates in two modes. In the color-blind mode, the meter treats the packet stream as uncolored. Any preset loss priorities are ignored. In the color-aware mode, the meter inspects the packet loss priority (PLP) field, which has been set by an upstream device as PLP high, medium-high, medium-low, or low; in other words, the PLP field has already been set by a behavior aggregate (BA) or multifield classifier. The marker changes the PLP of each incoming IP packet according to the results of the meter. For more information, see "Configuring Two-Rate Tricolor Marking" on page 17.

This chapter emphasizes configuration and use of TCM policers. For more information about configuring and using two-color policers ("policers"), see the Traffic Policers Configuration Guide.

Single-rate TCM is so called because traffic is policed according to one rate—the CBR—and two burst sizes: the CBS and EBS. The CBS specifies the usual burst size in bytes and the EBS specifies the maximum burst size in bytes for packets that are admitted to the network. The EBS is greater than or equal to the CBS, and neither can be 0. As each packet enters the network, its bytes are counted. Packets that do not exceed the CBS are marked low PLP. Packets that exceed the CBS but are below the EBS are marked medium-high PLP. Packets that exceed the EBS are marked high PLP.

Two-rate TCM is so called because traffic is policed according to two rates: the CIR and the PIR. The PIR is greater than or equal to the CIR. The CIR specifies the average rate at which bits are admitted to the network and the PIR specifies the maximum rate at which bits are admitted to the network. As each packet enters the network, its bits are counted. Bits in packets that do not exceed the CIR have their packets marked low PLP. Bits in packets that exceed the CIR but are below the PIR have their packets marked medium-high PLP. Bits in packets that exceed the PIR have their packets marked high PLP.

For information about how to use marking policers with BA and multifield classifiers, see "Using BA Classifiers to Set PLP" on page 27 and "Using Multifield Classifiers to Set PLP" on page 28.

Tricolor Marking Limitations

Tricolor Marking (TCM) has some limitations that must be kept in mind during configuration and operation.

The following limitations apply to TCM:

- When you enable TCM on a 10-port Gigabit Ethernet PIC or a 10-Gigabit Ethernet PIC, for queues 6 and 7 only, the output of the **show interfaces queue *interface-name*** command does not display the number of queued bytes and packets, or the number of bytes and packets dropped due to RED. If you do not configure tricolor marking on the interface, these statistics are available for all queues.
- When you enable TCM, Transmission Control Protocol (TCP)-based configurations for drop profiles are rejected. In other words, you cannot include the **protocol** statement at the **[edit class-of-service schedulers *scheduler-name* drop-profile-map]** hierarchy level. The result is that drop profiles are applied to packets with the specified PLP and any protocol type.
- On Gigabit Ethernet IQ PICs, for IEEE 802.1 rewrite rules, only two loss priorities are supported. Exiting packets with medium-high loss priority are treated as high, and packets with medium-low loss priority are treated as low. In other words rewrite rules corresponding to high and low apply instead of those corresponding to medium-high and medium-low. For IQ PICs, you can only configure one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule.
- When some PICs with Frame Relay encapsulation mark a packet with high loss priority, the packet is treated as having medium-high loss priority on M320 Multiservice Edge

Routers and T Series Core Routers with Enhanced II FPCs and T640 Core Routers with Enhanced Scaling FPC4.

- In a single firewall filter term, you cannot configure both the **loss-priority** action modifier and the **three-color-policer** action modifier. These statements are mutually exclusive.

Policer Support for Aggregated Ethernet Bundle Overview

Aggregated interfaces support single-rate policers, three-color marking policers, two-rate three-color marking policers, hierarchical policers, and percentage-based policers. By default, policer bandwidth and burst-size applied on aggregated bundles is not matched to the user-configured bandwidth and burst-size.

You can configure interface-specific policers applied on an aggregated Ethernet bundle to match the effective bandwidth and burst-size to user-configured values. The **shared-bandwidth-policer** statement is required to achieve this match behavior.

This capability applies to all interface-specific policers of the following types: single-rate policers, single-rate three-color marking policers, two-rate three-color marking policers, and hierarchical policers. Percentage-based policers match the bandwidth to the user-configured values by default, and do not require shared-bandwidth-policer configuration. The **shared-bandwidth-policer** statement causes a split in burst-size for percentage-based policers.

NOTE: This feature is supported on the following platforms: T Series routers, M120, M10i, M7i (CFEB-E only), M320 (SFPC only), MX240, MX480, and MX960 (DPC only),and EX Series switches.

The following usage scenarios are supported:

- Interface policers used by the following configuration:

  [edit] interfaces (ae*X* | as*X*) unit *unit-num* family *family* policer [input | output | arp]

- Policers and three-color policers (both single-rate three-color marking and two-rate three-color marking) used inside interface-specific filters; that is, filters that have an interface-specific keyword and are used by the following configuration:

  [edit] interfaces (ae*X* | as*X*) unit *unit-num* family *family* filter [input | output]

- Common-edge service filters, which are derived from CLI-configured filters and thus inherit interface-specific properties. All policers and three-color policers used by these filters are also affected.

The following usage scenarios are not supported:

- Policers and three-color policers used inside filters that are not interface specific; such a filter is meant to be shared across multiple interfaces.
- Any implicit policers or policers that are part of implicit filters; for example, the default ARP policer applied to an aggregate Ethernet interface. Such a policer is meant to be shared across multiple interfaces.

- Prefix-specific action policers.

To configure this feature, include the **shared-bandwidth-policer** statement at the following hierarchy levels: **[edit firewall policer** *policer-name***]**, **[edit firewall three-color-policer** *policer-name***]**, or **[edit firewall hierarchical-policer** *policer-name***]**.

Related Documentation

- shared-bandwidth-policer on page 71

PART 2

Configuration

CHAPTER 2

Configuration Tasks for Tricolor Marking Policers

Configuring Tricolor Marking

You configure marking policers by defining the policer and multiple levels of PLP for classifiers, rewrite rules, random early detection (RED) drop profiles, and firewall filters. To configure marking policers, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
tri-color;
classifiers {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) classifier-name {
      import classifier-name | default);
      forwarding-class class-name {
         loss-priority (low | medium-low | medium-high | high) code-points [ aliases ]
            [ bit-patterns ];
      }
   }
}
rewrite-rules {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) rewrite-name {
      import (rewrite-name | default);
      forwarding-class class-name {
         loss-priority (low | medium-low | medium-high | high) code-point (aliases |
            bit-patterns;
      }
   }
```

```
}
schedulers {
  scheduler-name {
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high)protocol
      any drop-profile profile-name;
  }
}

[edit firewall]
policer name {
  then loss-priority (low | medium-low | medium-high | high);
}
three-color-policer policer-name {
  action {
    loss-priority high then discard; # Only for IQ2 PICs
  }
  logical-interface-policer;
  single-rate {
    (color-aware | color-blind);
    committed-information-rate bps;
    committed-burst-size bytes;
    excess-burst-size bytes;
  }
  two-rate {
    (color-aware | color-blind);
    committed-information-rate bps;
    committed-burst-size bytes;
    peak-information-rate bps;
    peak-burst-size bytes;
  }
}
filter filter-name {
  <family family> {
    term rule-name {
      then {
        three-color-policer (single-rate | two-rate) policer-name;
      }
    }
  }
}
```

Configuring Single-Rate Tricolor Marking

With TCM, you can configure traffic policing according to two separate modes—color-blind and color-aware. In color-blind mode, the current PLP value is ignored. In color-aware mode, the current PLP values are considered by the policer and can only be increased.

- Configuring Color-Blind Mode for Single-Rate Tricolor Marking on page 15
- Configuring Color-Aware Mode for Single-Rate Tricolor Marking on page 15

Configuring Color-Blind Mode for Single-Rate Tricolor Marking

All packets are evaluated by the CBS. If a packet exceeds the CBS, it is evaluated by the EBS. In color-blind mode, the policer supports three loss priorities only: low, medium-high, and high.

In color-blind mode, packets that exceed the CBS but are below the EBS are marked yellow (medium-high). Packets that exceed the EBS are marked red (high), as shown in Table 3 on page 15.

Table 3: Color-Blind Mode TCM Color-to-PLP Mapping

Color	PLP	Meaning
Green	**low**	Packet does not exceed the CBS.
Yellow	**medium-high**	Packet exceeds the CBS but does not exceed the EBS.
Red	**high**	Packet exceeds the EBS.

If you are using color-blind mode and you wish to configure an output policer that marks packets to have medium-low loss priority, you must configure a policer at the **[edit firewall policer *policer-name*]** hierarchy level. For example:

```
firewall {
   policer 4PLP {
      if-exceeding {
         bandwidth-limit 40k;
         burst-size-limit 4k;
      }
      then loss-priority medium-low;
   }
}
```

Apply this policer at one or both of the following hierarchy levels:

- **[edit firewall family *family* filter *filter-name* term *rule-name* then policer *policer-name*]**
- **[edit interfaces *interface-name* unit *logical-unit-number* family *family* filter *filter-name*]**

Configuring Color-Aware Mode for Single-Rate Tricolor Marking

In color-aware mode, the metering treatment the packet receives depends on its classification. Metering can increase a packet's preassigned PLP, but cannot decrease it, as shown in Table 4 on page 16.

Table 4: Color-Aware Mode TCM PLP Mapping

Incoming PLP	Packet Metered Against	Possible Cases	Outgoing PLP
low	CBS and EBS	Packet does not exceed the CBS.	**low**
		Packet exceeds the CBS but not the EBS.	**medium-high**
		Packet exceeds the EBS.	**high**
medium-low	EBS only	Packet does not exceed the CBS.	**medium-low**
		Packet does not exceed the EBS.	**medium-low**
		Packet exceeds the EBS.	**high**
medium-high	EBS only	Packet does not exceed the CBS.	**medium-high**
		Packet does not exceed the EBS.	**medium-high**
		Packet exceeds the EBS.	**high**
high	Not metered by the policer.	All cases.	**high**

The following sections describe single-rate color-aware PLP mapping in more detail.

Effect on Low PLP of Single-Rate Policer

Packets belonging to the green class have already been marked by a classifier with low PLP. The marking policer can leave the packet's PLP unchanged or increase the PLP to medium-high or high. Therefore, these packets are metered against both the CBS and the EBS.

For example, if a BA or multifield classifier marks a packet with low PLP according to the type-of-service (ToS) bits in the IP header, and the two-rate TCM policer is in color-aware mode, the output loss priority is as follows:

- If the rate of traffic flow is less than the CBS, packets remain marked as low PLP.
- If the rate of traffic flow is greater than the CBS but less than the EBS, some of the packets are marked as medium-high PLP, and some of the packets remain marked as low PLP.
- If the rate of traffic flow is greater than the EBS, some of the packets are marked as high PLP, and some of the packets remain marked as low PLP.

Effect on Medium-Low PLP of Single-Rate Policer

Packets belonging to the yellow class have already been marked by a classifier with medium-low or medium-high PLP. The marking policer can leave the packet's PLP

unchanged or increase the PLP to high. Therefore, these packets are metered against the EBS only.

For example, if a BA or multifield classifier marks a packet with medium-low PLP according to the ToS bits in the IP header, and the two-rate TCM policer is in color-aware mode, the output loss priority is as follows:

- If the rate of traffic flow is less than the CBS, packets remain marked as medium-low PLP.
- If the rate of traffic flow is greater than the CBS but less than the EBS, packets remain marked as medium-low PLP.
- If the rate of traffic flow is greater than the EBS, some of the packets are marked as high PLP, and some of the packets remain marked as medium-low PLP.

Effect on Medium-High PLP of Single-Rate Policer

Packets belonging to the yellow class have already been marked by a classifier with medium-low or medium-high PLP. The marking policer can leave the packet's PLP unchanged or increase the PLP to high. Therefore, these packets are metered against the EBS only.

For example, if a BA or multifield classifier marks a packet with medium-high PLP according to the ToS bits in the IP header, and the two-rate TCM policer is in color-aware mode, the output loss priority is as follows:

- If the rate of traffic flow is less than the CBS, packets remain marked as medium-high PLP.
- If the rate of traffic flow is greater than the CBS but less than the EBS, packets remain marked as medium-high PLP.
- If the rate of traffic flow is greater than the EBS, some of the packets are marked as high PLP, and some of the packets remain marked as medium-high PLP.

Effect on High PLP of Single-Rate Policer

Packets belonging to the red class have already been marked by a classifier with high PLP. The marking policer can only leave the packet's PLP unchanged. Therefore, these packets are not metered against the CBS or the EBS and all the packets remain marked as high PLP.

Configuring Two-Rate Tricolor Marking

With TCM, you can configure traffic policing according to two separate modes—color-blind and color-aware. In color-blind mode, the current PLP value is ignored. In color-aware mode, the current PLP values are considered by the policer and can only be increased.

- Configuring Color-Blind Mode for Two-Rate Tricolor Marking on page 18
- Configuring Color-Aware Mode for Two-Rate Tricolor Marking on page 18

Configuring Color-Blind Mode for Two-Rate Tricolor Marking

All packets are evaluated by the CIR. If a packet exceeds the CIR, it is evaluated by the PIR. In color-blind mode, the policer supports three loss priorities only: low, medium-high, and high.

In color-blind mode, packets that exceed the CIR but are below the PIR are marked yellow (medium-high). Packets that exceed the PIR are marked red (high), as shown in Table 5 on page 18.

Table 5: Color-Blind Mode TCM Color-to-PLP Mapping

Color	PLP	Meaning
Green	**low**	Packet does not exceed the CIR.
Yellow	**medium-high**	Packet exceeds the CIR but does not exceed the PIR.
Red	**high**	Packet exceeds the PIR.

If you are using color-blind mode and you wish to configure an output policer that marks packets to have medium-low loss priority, you must configure a policer at the **[edit firewall policer *policer-name*]** hierarchy level. For example:

```
firewall {
   policer 4PLP {
      if-exceeding {
         bandwidth-limit 40k;
         burst-size-limit 4k;
      }
      then loss-priority medium-low;
   }
}
```

Apply this policer at one or both of the following hierarchy levels:

- **[edit firewall family *family* filer *filter-name* term *rule-name* then policer *policer-name*]**
- **[edit interfaces *interface-name* unit *logical-unit-number* family *family* filter *filter-name*]**

Configuring Color-Aware Mode for Two-Rate Tricolor Marking

In color-aware mode, the metering treatment the packet receives depends on its classification. Metering can increase a packet's preassigned PLP, but cannot decrease it, as shown in Table 6 on page 19.

Table 6: Color-Aware Mode TCM Mapping

Incoming PLP	Packet Metered Against	Possible Cases	Outgoing PLP
low	CIR and PIR	Packet does not exceed the CIR.	**low**
		Packet exceeds the CIR but not the PIR.	**medium-high**
		Packet exceeds the PIR.	**high**
medium-low	PIR only	Packet does not exceed the CIR.	**medium-low**
		Packet does not exceed the PIR.	**medium-low**
		Packet exceeds the PIR.	**high**
medium-high	PIR only	Packet does not exceed the CIR.	**medium-high**
		Packet does not exceed the PIR.	**medium-high**
		Packet exceeds the PIR.	**high**
high	Not metered by the policer.	All cases.	**high**

The following sections describe color-aware two-rate PLP mapping in more detail.

Effect on Low PLP of Two-Rate Policer

Packets belonging to the green class have already been marked by a classifier with low PLP. The marking policer can leave the packet's PLP unchanged or increase the PLP to medium-high or high. Therefore, these packets are metered against both the CIR and the PIR.

For example, if a BA or multifield classifier marks a packet with low PLP according to the ToS bits in the IP header, and the two-rate TCM policer is in color-aware mode, the output loss priority is as follows:

- If the rate of traffic flow is less than the CIR, packets remain marked as low PLP.
- If the rate of traffic flow is greater than the CIR but less than the PIR, some of the packets are marked as medium-high PLP, and some of the packets remain marked as low PLP.
- If the rate of traffic flow is greater than the PIR, some of the packets are marked as high PLP, and some of the packets remain marked as low PLP.

Effect on Medium-Low PLP of Two-Rate Policer

Packets belonging to the yellow class have already been marked by a classifier with medium-low or medium-high PLP. The marking policer can leave the packet's PLP

unchanged or increase the PLP to high. Therefore, these packets are metered against the PIR only.

For example, if a BA or multifield classifier marks a packet with medium-low PLP according to the ToS bits in the IP header, and the two-rate TCM policer is in color-aware mode, the output loss priority is as follows:

- If the rate of traffic flow is less than the CIR, packets remain marked as medium-low PLP.
- If the rate of traffic flow is greater than the CIR/CBS but less than the PIR, packets remain marked as medium-low PLP.
- If the rate of traffic flow is greater than the PIR, some of the packets are marked as high PLP, and some of the packets remain marked as medium-low PLP.

Effect on Medium-High PLP of Two-Rate Policer

Packets belonging to the yellow class have already been marked by a classifier with medium-low or medium-high PLP. The marking policer can leave the packet's PLP unchanged or increase the PLP to high. Therefore, these packets are metered against the PIR only.

For example, if a BA or multifield classifier marks a packet with medium-high PLP according to the ToS bits in the IP header, and the two-rate TCM policer is in color-aware mode, the output loss priority is as follows:

- If the rate of traffic flow is less than the CIR, packets remain marked as medium-high PLP.
- If the rate of traffic flow is greater than the CIR but less than the PIR, packets remain marked as medium-high PLP.
- If the rate of traffic flow is greater than the PIR, some of the packets are marked as high PLP, and some of the packets remain marked as medium-high PLP.

Effect on High PLP of Two-Rate Policer

Packets belonging to the red class have already been marked by a classifier with high PLP. The marking policer can only leave the packet's PLP unchanged. Therefore, these packets are not metered against the CIR or the PIR and all the packets remain marked as high PLP.

Enabling Tricolor Marking

By default, TCM is enabled on M120, MX Series, and T4000 routers, and EX Series switches. To enable TCM on other routers, include the **tri-color** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
tri-color;
```

This statement is necessary on the following routers:

- M320 and T Series routers with Enhanced II FPCs
- T640 routers with Enhanced Scaling FPC4s

If you do not include this statement in the configuration on platforms that require it, you cannot configure medium-low or medium-high PLP for classifiers, rewrite rules, drop profiles, or firewall filters.

Configuring Tricolor Marking Policers

A tricolor marking policer polices traffic on the basis of metering rates, including the CIR, the PIR, their associated burst sizes, and any policing actions configured for the traffic. To configure a tricolor marking policer, include the following statements at the **[edit firewall]** hierarchy level:

```
[edit firewall]
three-color-policer name {
  action {
    loss-priority high then discard; # Only for IQ2 PICs
  }
  logical-interface-policer;
  single-rate {
    (color-aware | color-blind);
    committed-information-rate bps;
    committed-burst-size bytes;
    excess-burst-size bytes;
  }
  two-rate {
    (color-aware | color-blind);
    committed-information-rate bps;
    committed-burst-size bytes;
    peak-information-rate bps;
    peak-burst-size bytes;
  }
}
```

You can configure a tricolor policer to discard high loss priority traffic on a logical interface in the ingress or egress direction. To configure a policer on a logical interface using tricolor marking policing to discard high loss priority traffic, include the **logical-interface-policer** statement and **action** statement.

In all cases, the range of allowable bits-per-second or byte values is 1500 to 100,000,000,000. You can specify the values for bps and bytes either as complete decimal numbers or as decimal numbers followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

The color-aware policer implicitly marks packets into four loss priority categories:

- Low
- Medium-low

- Medium-high
- High

The color-blind policer implicitly marks packets into three loss priority categories:

- Low
- Medium-high
- High

Table 7 on page 22 describes all the configurable TCM statements.

Table 7: Tricolor Marking Policer Statements

Statement	Meaning	Configurable Values
single-rate	Marking is based on the CIR, CBS, and EBS.	–
two-rate	Marking is based on the CIR, PIR, and rated burst sizes.	–
color-aware	Metering depends on the packet's preclassification. Metering can increase a packet's assigned PLP, but cannot decrease it.	–
color-blind	All packets are evaluated by the CIR or CBS. If a packet exceeds the CIR or CBS, it is evaluated by the PIR or EBS.	–
committed-information-rate	Guaranteed bandwidth under normal line conditions and the average rate up to which packets are marked green.	1500 through 100,000,000,000 bps
committed-burst-size	Maximum number of bytes allowed for incoming packets to burst above the CIR, but still be marked green.	1500 through 100,000,000,000 bytes
excess-burst-size	Maximum number of bytes allowed for incoming packets to burst above the CIR, but still be marked yellow.	1500 through 100,000,000,000 bytes
peak-information-rate	Maximum achievable rate. Packets that exceed the CIR but are below the PIR are marked yellow. Packets that exceed the PIR are marked red.	1500 through 100,000,000,000 bps
peak-burst-size	Maximum number of bytes allowed for incoming packets to burst above the PIR, but still be marked yellow.	1500 through 100,000,000,000 bytes

Applying Tricolor Marking Policers to Firewall Filters

To rate-limit traffic by applying a tricolor marking policer to a firewall filter, include the **three-color-policer** statement:

```
three-color-policer {
   (single-rate | two-rate) policer-name;
}
```

You can include this statement at the following hierarchy levels:

- **[edit firewall family *family* filter *filter-name* term *rule-name* then]**
- **[edit firewall filter *filter-name* term *rule-name* then]**

In the **family** statement, the protocol family can be **any**, **ccc**, **inet**, **inet6**, **mpls**, or **vpls**.

You must identify the referenced policer as a **single-rate** or **two-rate** policer, and this statement must match the configured TCM policer. Otherwise, an error message appears in the configuration listing.

For example, if you configure **srTCM** as a single-rate TCM policer and try to apply it as a two-rate policer, the following message appears:

```
[edit firewall]
user@host# show three-color-policer srTCM
single-rate {
   color-aware;
   ...
}
user@host# show filter TESTER
term A {
   then {
      three-color-policer {
         ##
         ## Warning: Referenced two-rate policer does not exist
         ##
         two-rate srTCM;
      }
   }
}
```

Example: Applying a Two-Rate Tricolor Marking Policer to a Firewall Filter

Apply the **trtcm1-cb** policer to a firewall filter:

```
firewall {
   three-color-policer trtcm1-cb {  # Configure the trtcm1-cb policer.
   two-rate {
      color-blind;
      committed-information-rate 1048576;
      committed-burst-size 65536;
      peak-information-rate 10485760;
      peak-burst-size 131072;
   }
}
filter fil {   # Configure the fil firewall filter, applying the trtcm1-cb policer.
term default {
   then {
      three-color-policer {
         two-rate trtcm1-cb;
      }
   }
}
```

Related Documentation

- Firewall Filters Configuration Guide

Applying Firewall Filter Tricolor Marking Policers to Interfaces

To apply a tricolor marking policer to an interface, you must reference the filter name in the interface configuration. To do this, include the **filter** statement:

```
filter {
  input filter-name;
  output filter-name;
}
```

You can include these statements at the following hierarchy levels:

- **[edit interfaces *interface-name* unit *logical-unit-number* family *family*]**
- **[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family*]**

The filter name that you reference should have an attached tricolor marking policer, as shown in "Applying Tricolor Marking Policers to Firewall Filters" on page 22.

Example: Applying a Single-Rate Tricolor Marking Policer to an Interface

Apply the **trtcm1-cb** policer to an interface:

```
firewall {
  three-color-policer srtcm1 {    # Configure the srtcm1-cb policer.
  single-rate {
    color-blind;
    committed-information-rate 1048576;
    committed-burst-size 65536;
    excess-burst-size 131072;
  }
}
filter fil {    # Configure the fil firewall filter, applying the srtcm1-cb policer.
term default {
  then {
    three-color-policer {
      single-rate srtcm1-cb;    # The TCM policer must be single-rate.
    }
  }
}
interfaces {    # Configure the interface, which attaches the fil firewall filter.
ge-1/0/0 {
  unit 0 {
    family inet {
      filter {
        input fil;
      }
    }
  }
}
```

Applying Layer 2 Policers to Gigabit Ethernet Interfaces

To rate-limit traffic by applying a policer to a Gigabit Ethernet interface (or a 10-Gigabit Ethernet interface [**xe-*fpc/pic/port***]), include the **layer2-policer** statement with the direction, type, and name of the policer:

```
[edit interfaces ge-fpc/pic/port unit 0]
layer2-policer {
  input-policer policer-name;
  input-three-color policer-name;
  output-policer policer-name;
  output-three-color policer-name;
}
```

The direction (input or output) and type (policer or three-color) are combined into one statement and the policer named must be properly configured.

One input or output policer of either type can be configured on the interface.

Examples: Applying Layer 2 Policers to a Gigabit Ethernet Interface

Apply color-blind and color-aware two-rate TCM policers as input and output policers to a Gigabit Ethernet interface:

```
ge-1/0/0 {
  unit 0
  layer2-policer {
    input-three-color trTCM1-cb;   # Apply the trTCM1-color-blind policer.
    output-three-color trTCM1-ca;   # Apply the trTCM1-color-aware policer.
  }
}
```

Apply two-level and color-blind single-rate TCM policers as input and output policers to a Gigabit Ethernet interface:

```
ge-1/0/0 {
  unit 1
  layer2-policer {
    input-policer two-color-policer;   # Apply a two-color policer.
    output-three-color srTCM2-cb;   # Apply the srTCM1-color-blind policer.
  }
}
```

Apply a color-aware single-rate TCM policer as output policer on a Gigabit Ethernet interface:

```
ge-1/0/0 {
  unit 2
  layer2-policer {
    output-three-color srTCM3-ca {   # Apply the srTCM3-color-aware policer.
  }
}
```

CHAPTER 3

Configuration Tasks for Packet Loss Priority

Using BA Classifiers to Set PLP

Behavior aggregate (BA) classifiers take action on incoming packets. When TCM is enabled, Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers support four classifier PLP designations: **low**, **medium-low**, **medium-high**, and **high**. To configure the PLP for a classifier, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
classifiers {
  (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) classifier-name {
    import (classifier-name | default);
    forwarding-class class-name {
      loss-priority (low | medium-low | medium-high | high) code-points [ aliases ]
        [ bit-patterns ];
    }
  }
}
```

The inputs for a classifier are the CoS values. The outputs for a classifier are the forwarding class and the loss priority (PLP). A classifier sets the forwarding class and the PLP for each packet entering the interface with a specific set of CoS values.

For example, in the following configuration, the **assured-forwarding** forwarding class and **medium-low** PLP are assigned to all packets entering the interface with the **101110** CoS values:

```
class-of-service {
  classifiers {
    dscp dscp-cl {
      forwarding-class assured-forwarding {
        loss-priority medium-low {
```

```
            code-points 101110;
        }
      }
    }
  }
}
```

To use this classifier, you must configure the settings for the **assured-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* assured-forwarding]** hierarchy level. For more information, see Overview of Forwarding Classes.

Using Multifield Classifiers to Set PLP

Multifield classifiers take action on incoming or outgoing packets, depending whether the firewall rule is applied as an input filter or an output filter. When TCM is enabled, Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers support four multifield classifier PLP designations: **low**, **medium-low**, **medium-high**, and **high**.

To configure the PLP for a multifield classifier, include the **loss-priority** statement in a policer or firewall filter that you configure at the at the **[edit firewall]** hierarchy level:

```
[edit firewall]
family family-name {
  filter filter-name {
    term term-name {
      from {
        match-conditions;
      }
      then {
        loss-priority (low | medium-low | medium-high | high);
        forwarding-class class-name;
      }
    }
  }
}
```

The inputs (match conditions) for a multifield classifier are one or more of the six packet header fields: destination address, source address, IP protocol, source port, destination port, and DSCP. The outputs for a multifield classifier are the forwarding class and the loss priority (PLP). In other words, a multifield classifier sets the forwarding class and the PLP for each packet entering or exiting the interface with a specific destination address, source address, IP protocol, source port, destination port, or DSCP.

For example, in the following configuration, the forwarding class **expedited-forwarding** and PLP **medium-high** are assigned to all IPv4 packets with the **10.1.1.0/24** or **10.1.2.0/24** source address:

```
firewall {
  family inet {
    filter classify-customers {
      term isp1-customers {
        from {
          source-address 10.1.1.0/24;
```

```
            source-address 10.1.2.0/24;
          }
          then {
            loss-priority medium-high;
            forwarding-class expedited-forwarding;
          }
        }
      }
    }
  }
```

To use this classifier, you must configure the settings for the **expedited-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* expedited-forwarding]** hierarchy level. For more information, see Overview of Forwarding Classes.

Configuring PLP for Drop-Profile Maps

RED drop profiles take action on outgoing packets. When TCM is enabled, M320 and T Series routers support four drop-profile map PLP designations: **low**, **medium-low**, **medium-high**, and **high**.

To configure the PLP for the drop-profile map, include the **schedulers** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
schedulers {
  scheduler-name {
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high)protocol
      any drop-profile profile-name;
  }
}
```

When you configure TCM, the drop-profile map's protocol type must be **any**.

The inputs for a drop-profile map are the loss priority and the protocol type. The output for a drop-profile map is the drop profile name. In other words, the map sets the drop profile for each packet with a specific PLP and protocol type exiting the interface.

For example, in the following configuration, the **dp** drop profile is assigned to all packets exiting the interface with a medium-low PLP and belonging to any protocol:

```
class-of-service {
  schedulers {
    af {
      drop-profile-map loss-priority medium-low protocol any drop-profile dp;
    }
  }
}
```

To use this drop-profile map, you must configure the settings for the **dp** drop profile at the **[edit class-of-service drop-profiles dp]** hierarchy level. For more information, see RED Drop Profiles Overview.

Configuring Rewrite Rules Based on PLP

Rewrite rules take action on outgoing packets. When TCM is enabled, M320 and T Series routers support four rewrite PLP designations: **low**, **medium-low**, **medium-high**, and **high**. To configure the PLP for a rewrite rule, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
rewrite-rules {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) rewrite-name {
      import (rewrite-name | default);
      forwarding-class class-name {
         loss-priority (low | medium-low | medium-high | high) code-point (alias | bits);
      }
   }
}
```

The inputs for a rewrite rule are the forwarding class and the loss priority (PLP). The output for a rewrite rule are the CoS values. In other words, a rewrite rule sets the CoS values for each packet exiting the interface with a specific forwarding class and PLP.

For example, if you configure the following, the **000000** CoS values are assigned to all packets exiting the interface with the **assured-forwarding** forwarding class and **medium-high** PLP:

```
class-of-service {
   rewrite-rules {
      dscp dscp-rw {
         forwarding-class assured-forwarding {
            loss-priority medium-high code-point 000000;
         }
      }
   }
}
```

To use this classifier, you must configure the settings for the **assured-forwarding** forwarding class at the **[edit class-of-service forwarding-classes queue *queue-number* assured-forwarding]** hierarchy level. For more information, see Overview of Forwarding Classes.

CHAPTER 4

Configuration Statements for Tricolor Marking Policers

[edit class-of-service] Hierarchy Level

```
class-of-service {
  classifiers {
    type classifier-name {
      forwarding-class class-name {
        loss-priority (high | low | medium-high | medium-low) code-points [ aliases bits ];
      }
      import (classifier-name | default);
    }
  }
  code-point-aliases {
    (dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) {
      alias-name bits;
    }
  }
  drop-profiles {
    profile-name {
      fill-level percentage drop-probability percentage;
      interpolate {
        drop-probability value;
        fill-level value;
      }
    }
  }
  fabric {
    scheduler-map {
      priority (high | low) scheduler scheduler-name;
    }
  }
  forwarding-class-map {
    map-name {
      class class-name queue-num queue-number <restricted-queue queue-number>;
    }
```

```
}
forwarding-classes {
  class class-name policing-priority (normal | premium) queue-num queue-number
    priority (high | low);
  queue queue-number class-name policing-priority (normal | premium) priority (high |
    low);
}
forwarding-policy {
  class class-name {
    classification-override {
      forwarding-class class-name;
    }
  }
  next-hop-map map-name {
    forwarding-class class-name {
      discard;
      lsp-next-hop [ lsp-regular-expressions ];
      next-hop [ next-hop-names ];
      non-lsp-next-hop;
    }
  }
}
fragmentation-maps {
  map-name {
    forwarding-class class-name {
      drop-timeout milliseconds;
      fragment-threshold bytes;
      multilink-class number;
      no-fragmentation;
    }
  }
}
host-outbound-traffic {
  dscp-code-point value;
  forwarding-class class-name;
  ieee-802.1 {
    default value;
    rewrite-rules;
  }
  tcp {
    raise-internet-control-priority;
  }
}
interfaces {
  ... the interfaces subhierarchy appears after the main [edit class-of-service] hierarchy
    ...
}
}
restricted-queues {
  forwarding-class class-name queue-number;
}
rewrite-rules {
  (dscp | dscp-ipv6 | exp |frame-relay-de | ieee-802.1 | ieee-802.1ad | inet-precedence)
    rewrite-rule {
    forwarding-class class-name {
      loss-priority level code-point (alias | bits);
```

```
        }
        import (rewrite-rule | default);
      }
    }
    routing-instances routing-instance-name {
      classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default | encapsulated | vlan-tag (inner | outer));
      }
    }
    scheduler-maps {
      map-name {
        forwarding-class class-name scheduler scheduler-name;
      }
    }
    schedulers {
      scheduler-name {
        adjust-minimum value;
        adjust-percent value;
        buffer-size (exact | percent percentage | remainder);
        drop-profile-map loss-priority (any | high | low | medium-high | medium-low)
          protocol any;
        excess-priority (high | low | medium-high | medium-low);
        excess-rate (percent percentage | proportion proportion);
        priority (high | low | medium-high | medium-low | strict-high);
        shaping-rate (bps | percent percentage | burst-size size);
        transmit-rate (bps | percent percentage | remainder) <exact | rate-limit>;
      }
    }
    traceoptions {
      file <files number> <match regular-expression> <size maximum-file-size>
        <world-readable | no-world-readable>;
      flag flag;
      no-remote-trace;
    }
    traffic-control-profiles {
      profile-name {
        adjust-minimum rate;
        delay-buffer-rate (bps | cps cps | percent percentage);
        excess-rate (percent percentage | proportion value);
        guaranteed-rate (bps | percent percentage) <burst-size bytes>;
        overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
        scheduler-map map-name;
        shaping-rate (bps | percent percentage) <burst-size bytes>;
      }
    }
    tri-color;
}

class-of-service {
  interfaces {
    interface-name {
      excess-bandwith-share (equal | proportional value);
      input-excess-bandwith-share (equal | proportional value);
```

```
    input-scheduler-map map-name;
    input-shaping-rate bps;
    input-traffic-control-profile profile-name;
    output-forwarding-class-map map-name;
    output-traffic-control-profile profile-name;
    scheduler-map map-name;
    scheduler-map-chassis (map-name | derived);
    shaping-rate bps;
    unit (logical-unit-number | *){
      classifiers {
        dscp (classifier-name | default) {
          family [ inet mpls ];
        }
        dscp-ipv6 (classifier-name | default) {
          family [ inet mpls ];
        }
        exp (classifier-name | default);
        ieee-208.1 (classifier-name | default)  <vlan-tag (inner | outer)>;
        ieee-208.1ad (classifier-name | default);
        inet-precedence (classifier-name | default);
      }
      forwarding-class class-name;
      input-scheduler-map map-name;
      input-shaping-rate bps;
      input-traffic-control-profile profile-name shared-instance instance-name;
      loss-priority-maps {
        (map-name | default);
      }
      loss-priority-rewrites {
        (map-name | default);
      }
      output-forwarding-class-map map-name;
      output-traffic-control-profile profile-name shared-instance instance-name;
      rewrite-rules {
        dscp (rule-name | default) <protocol mpls>;
        dscp-ipv6 (rule-name | default);
        exp (rule-name | default) <protocol [ mpls-any | mpls-inet-both |
          mpls-inet-both-non-vpn ]>;
        exp-push-push-push default;
        exp-swap-push-push default;
        ieee-802.1 (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        ieee-802.1ad (rewrite-name | default) <vlan-tag (outer | outer-and-inner)>;
        inet-precedence (rewrite-name | default) <protocol mpls>;
      }
      scheduler-map map-name;
      shaping-rate bps;
      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-bandwith-share (equal | proportional value);
    input-excess-bandwith-share (equal | proportional value);
    input-traffic-control-profile profile-name;
    input-traffic-control-profile-remaining profile-name;
    internal-node;
```

```
            output-traffic-control-profile profile-name;
            output-traffic-control-profile-remaining profile-name;
        }
    }
}
```

Related Documentation

- Notational Conventions Used in Junos OS Configuration Hierarchies

classifiers (Definition)

Syntax

```
classifiers {
    type classifier-name {
        import (classifier-name | default);
        forwarding-class class-name {
            loss-priority level code-points [ aliases ] [ bit-patterns ];
        }
    }
}
```

Hierarchy Level

[edit class-of-service],
[edit class-of-service routing-instances *routing-instance-name*]

Release Information

Statement introduced before Junos OS Release 7.4.
ieee-802.1ad option introduced in Junos OS Release 9.2.

Description

Define a CoS behavior aggregate (BA) classifier for classifying packets. You can associate the classifier with a forwarding class or code-point mapping, and import a default classifier or one that is previously defined.

NOTE: The **[edit class-of-service routing-instances *routing-instance-name*]** hierarchy level and the **dscp-ipv6** and **ieee-802.1ad** classifier types are not supported on ACX Series routers.

Options

classifier-name—Name of the aggregate behavior classifier.

type—Traffic type: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ieee-802.1ad**, **inet-precedence**.

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
- Example: Configuring CoS for a PBB Network
- Configuring CoS on ACX Series Universal Access Routers

code-points

Syntax code-points ([*aliases*] | [*bit-patterns*]);

Hierarchy Level [edit class-of-service classifiers *type classifier-name* forwarding-class *class-name* loss-priority *level*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 8.5 for J Series devices.
Statement introduced in Junos OS Release 9.2 for SRX Series devices.

Description Specify one or more DSCP code-point aliases or bit sets for association with a forwarding class.

Options ***aliases***—Name of the DSCP alias.

bit-patterns—Value of the code-point bits, in six-bit binary form.

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
- Example: Configuring CoS for a PBB Network
- Example: Configuring Behavior Aggregate Classifiers
- Example: Configuring Forwarding Classes

drop-profile (Schedulers)

Syntax drop-profile *profile-name*;

Hierarchy Level [edit class-of-service schedulers *scheduler-name* drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol (any | non-tcp | tcp)]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Switches.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description Define drop profiles for RED. When a packet arrives, RED checks the queue fill level. If the fill level corresponds to a nonzero drop probability, the RED algorithm determines whether to drop the arriving packet.

Options *profile-name*—Name of the drop profile.

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

Related Documentation
- Configuring Drop Profile Maps for Schedulers
- RED Drop Profiles Overview

drop-profile-map (Schedulers)

Syntax drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol(any | non-tcp | tcp) drop-profile (Schedulers) *profile-name*;

Hierarchy Level [edit class-of-service schedulers *scheduler-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Switches.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description Define the loss-priority value for a drop profile.

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
- Default Schedulers Overview
- Configuring Drop Profile Maps for Schedulers

dscp (Multifield Classifier)

Syntax dscp [0 | *value*];

Hierarchy Level [edit firewall family *family-name* filter *filter-name* term *term-name* then]

Release Information Statement introduced in Junos OS Release 7.4.

Description For M320 and T Series routers, set the DSCP field of incoming or outgoing packets to **000000**. On the same packets, you can use a behavior aggregate (BA) classifier and a rewrite rule to rewrite the MPLS EXP field.

For MX Series routers with MPCs and EX Series switches, the DSCP field can be set from a numeric range.

For MX Series routers and EX Series switches, if you configure a firewall filter with a DSCP action or traffic-class action on a DPC, the commit does not fail, but the filter is not applied to the interface, a warning displays, and an entry is made in the syslog.

Options *value*—For MX Series routers with MPCs and EX Series switches, specify the field of incoming or outgoing packets in the range from **0** through **63**.

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

Related Documentation

- Applying Tricolor Marking Policers to Firewall Filters on page 22

dscp (Rewrite Rules)

Syntax dscp (*rewrite-name* | default);

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule.

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

default—The default mapping.

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

Related Documentation

- Configuring Rewrite Rules
- dscp-ipv6 (Class-of-Service) on page 40
- exp on page 41
- exp-push-push-push
- exp-swap-push-push
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 43
- ieee-802.1ad
- inet-precedence on page 45
- rewrite-rules (Definition) on page 48

dscp-ipv6 (Class-of-Service)

Syntax

```
dscp-ipv6 (rewrite-name | <default>) {
  protocol mpls
}
```

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

Release Information Statement introduced before Junos OS Release 7.4.

Description For IPv6 traffic, apply a DSCP rewrite rule.

Options *rewrite-name*—Name of a **rewrite-rules** mapping configured at the **[edit class-of-service rewrite-rules dscp-ipv6]** hierarchy level.

default— Default mapping.

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 Rewrite Rules
- dscp (Rewrite Rules) on page 39
- exp on page 41
- exp-push-push-push
- exp-swap-push-push
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 43
- ieee-802.1ad
- inet-precedence on page 45
- rewrite-rules (Definition) on page 48

exp

Syntax exp (*rewrite-name* | default) protocol *protocol-types*;

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

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS Release 12.2. for ACX series

Description Apply an MPLS experimental (EXP) rewrite rule.

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

default—The default mapping.

By default, IP precedence rewrite rules alter the first three bits on the type-of-service (ToS) byte while leaving the last three bits unchanged. This default behavior applies to rewrite rules you configure for MPLS packets with IPv4 payloads. You configure these types of rewrite rules by including the **mpls-inet-both** or **mpls-inet-both-non-vpn** option at the **[edit class-of-service interfaces interface *interface-name* unit *logical-unit-number* rewrite-rules exp *rewrite-rule-name* protocol]** hierarchy level. The IP precedence rewrite rules explanation does not apply to ACX Series Universal Access routers.

On interfaces configured on Modular Port Concentrators (MPCs) and Modular Interface Cards (MICs) on MX Series Ethernet Services Routers and EX Series switches, we highly recommend that you configure the **default** option when you configure a behavior aggregate (BA) classifier that does not include a specific rewrite rule for MPLS packets. Doing so ensures that MPLS exp value is rewritten according to the BA classifier rules configured for forwarding or packet loss priority. This does not apply to ACX Series Universal Access routers.

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

- Rewriting the EXP Bits of All Three Labels of an Outgoing Packet
- dscp (Rewrite Rules) on page 39
- dscp-ipv6 (Class-of-Service) on page 40
- exp-push-push-push
- exp-swap-push-push
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 43
- ieee-802.1ad
- inet-precedence on page 45

- rewrite-rules (Definition) on page 48

forwarding-class (BA Classifiers)

Syntax

```
forwarding-class class-name {
  loss-priority level code-points [ aliases ] [ bit-patterns ];
}
```

Hierarchy Level [edit class-of-service classifiers *type classifier-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define forwarding class name and option values.

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

- Defining Classifiers
- Example: Configuring CoS for a PBB Network

ieee-802.1 (Rewrite Rules on Logical Interface)

Syntax ieee-802.1 (*rewrite-name* | default) vlan-tag (outer | outer-and-inner);

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

Release Information Statement introduced before Junos OS Release 7.4.
vlan-tag statement introduced in Junos OS Release 8.1.

Description Apply an IEEE-802.1 rewrite rule. For IQ PICs, you can only configure one IEEE 802.1 rewrite rule on a physical port. All logical ports (units) on that physical port should apply the same IEEE 802.1 rewrite rule.

Options *rewrite-name*—Name of a **rewrite-rules** mapping configured at the **[edit class-of-service rewrite-rules ieee-802.1]** hierarchy level.

default—The default mapping.

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

Related Documentation

- Configuring Rewrite Rules
- Example: Configuring CoS for a PBB Network
- dscp (Rewrite Rules) on page 39
- dscp-ipv6 (Class-of-Service) on page 40
- exp on page 41
- exp-push-push-push
- exp-swap-push-push
- ieee-802.1ad
- inet-precedence on page 45
- rewrite-rules (Definition) on page 48

import (Classifiers)

Syntax import (*classifier-name* | default);

Hierarchy Level [edit class-of-service classifiers *type classifier-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify a default or previously defined classifier.

Options ***classifier-name***—Name of the classifier mapping configured at the **[edit class-of-service classifiers]** hierarchy level.

default—The default classifier mapping.

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

import (Rewrite Rules)

Syntax import (*rewrite-name* | default);

Hierarchy Level [edit class-of-service rewrite-rules *type rewrite-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Specify a default or previously defined **rewrite-rules** mapping to import.

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

default—The default **rewrite-rules** mapping.

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

Related Documentation
- Configuring Rewrite Rules

inet-precedence

Syntax inet-precedence (*rewrite-name* | default);

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

Release Information Statement introduced before Junos OS Release 7.4.

Description Apply a IPv4 precedence rewrite rule.

Options

rewrite-name—Name of a **rewrite-rules** mapping configured at the **[edit class-of-service rewrite-rules inet-precedence]** hierarchy level.

default—The default mapping. By default, IP precedence rewrite rules alter the first three bits on the type of service (ToS) byte while leaving the last three bits unchanged.

Required Privilege Level

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

Related Documentation

- Configuring Rewrite Rules
- dscp (Rewrite Rules) on page 39
- dscp-ipv6 (Class-of-Service) on page 40
- exp on page 41
- exp-push-push-push
- exp-swap-push-push
- ieee-802.1 (Rewrite Rules on Logical Interface) on page 43
- ieee-802.1ad
- rewrite-rules (Definition) on page 48

loss-priority (Scheduler Drop Profiles)

Syntax loss-priority (any | high | low | medium-high | medium-low);

Hierarchy Level [edit class-of-service schedulers *scheduler-name* drop-profile-map]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Switches.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description Specify a loss priority to which to apply a drop profile. The drop profile map sets the drop profile for a specific PLP and protocol type. The inputs for the map are the PLP designation and the protocol type. The output is the drop profile.

Options **any**—The drop profile applies to packets with any PLP.

NOTE: On ACX Series Routers, only the **any** option is supported when you configure the **non-tcp** option for protocol.

high—The drop profile applies to packets with high PLP.

low—The drop profile applies to packets with low PLP.

medium-high—The drop profile applies to packets with medium-high PLP.

medium-low—The drop profile applies to packets with medium-low PLP.

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

Related Documentation

- Default Schedulers Overview
- Configuring Drop Profile Maps for Schedulers
- Configuring Schedulers for Priority Scheduling
- Configuring Tricolor Marking on page 13
- protocol (Schedulers) on page 47

protocol (Schedulers)

Syntax protocol (any | non-tcp | tcp);

Hierarchy Level [edit class-of-service schedulers *scheduler-name* drop-profile-map]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series Packet Transport Switches.
Statement introduced in Junos OS Release 12.2 for ACX Series Routers.

Description Specify the protocol type for the specified scheduler.

Options **any**—Accept any protocol type.

non-tcp—(ACX Series Routers, M Series and T Series (except T4000) routers only) Accept any protocol type other than TCP/IP.

NOTE: On ACX Series Routers, when you configure the non-tcp option, only the any option is supported for loss-priority.

tcp—(ACX Series Routers, M Series and T Series (except T4000) routers only) Accept TCP/IP protocol type.

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

Related Documentation

- Configuring Schedulers

rewrite-rules (Definition)

Syntax

```
rewrite-rules {
  type rewrite-name{
    import (rewrite-name | default);
    forwarding-class class-name {
      loss-priority level code-point [ aliases ] [ bit-patterns ];
    }
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
ieee-802.1ad option introduced in Junos OS Release 9.2.

Description Specify a rewrite-rules mapping for the traffic that passes through all queues on the interface.

Options *rewrite-name*—Name of a **rewrite-rules** mapping.

type—Traffic type.

Values: **dscp**, **dscp-ipv6**, **exp**, **frame-relay-de** (J Series routers only), **ieee-802.1**, **ieee-802.1ad**, **inet-precedence**

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 Rewrite Rules
- Example: Configuring CoS for a PBB Network
- J Series router documentation

schedulers (Class of Service)

Syntax

```
schedulers {
  scheduler-name {
    adjust-minimum rate;
    adjust-percent percentage;
    buffer-size (seconds | percent percentage | remainder | temporal microseconds);
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol
      (any | non-tcp | tcp) drop-profile profile-name;
    excess-priority [ low | medium-low | medium-high | high | none];
    excess-rate (percent percentage | proportion value);
    priority priority-level;
    shaping-rate (percent percentage | rate);
    transmit-rate (percent percentage | rate | remainder) <exact | rate-limit>;
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced in Junos OS Release 12.1X48 for PTX Series switches.

Description Specify the scheduler name and parameter values.

Options ***scheduler-name***—Name of the scheduler to be configured.

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

- Schedulers Overview
- Default Schedulers Overview
- Configuring Schedulers
- Configuring a Scheduler
- Example: Configuring CoS for a PBB Network

tri-color

Syntax tri-color;

Hierarchy Level [edit class-of-service]

Release Information Statement introduced in Junos OS Release 7.4.

Description For IPv4 packets on M320, MX Series, T Series routers with Enhanced II Flexible PIC Concentrators (FPCs), and EX Series switches, enable two-rate tricolor marking (TCM), as defined in RFC 2698.

Default If you do not include this statement, tricolor marking is not enabled and the medium packet loss priority (PLP) is not configurable.

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

Related Documentation

- Configuring Tricolor Marking on page 13

[edit firewall] Hierarchy Level

Several statements in the **[edit firewall]** hierarchy are valid at numerous locations within the hierarchy. To make the complete hierarchy easier to read, the repeated statements are listed in the following sections, which are referenced at the appropriate locations in "Complete [edit firewall] Hierarchy" on page 55.

- Common Firewall Actions on page 50
- Common IP Firewall Actions on page 51
- Common IPv4 Firewall Actions on page 51
- Common IP Firewall Match Conditions on page 52
- Common IPv4 Firewall Match Conditions on page 53
- Common Layer 2 Firewall Match Conditions on page 53
- Complete [edit firewall] Hierarchy on page 55

Common Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit firewall] Hierarchy" on page 55 instead of the statements being repeated.

- **[edit firewall family (any | ccc | ethernet-switching | inet | inet6 | mpls | vpls) filter *filter-name* term *term-name* then]**
- **[edit firewall filter *filter-name* term *term-name* then]**

The common firewall actions are as follows:

```
count counter-name;
forwarding-class class-name;
loss-priority (high | low | medium-high | medium-low);
next term;
policer policer-name;
three-color-policer policer-name {
   (single-rate single-rate-policer-name | two-rate two-rate-policer-name);
}
```

Common IP Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit firewall] Hierarchy" on page 55 instead of the statements being repeated.

- **[edit firewall family inet filter *filter-name* term *term-name* then]**
- **[edit firewall family inet6 filter *filter-name* term *term-name* then]**
- **[edit firewall filter *filter-name* term *term-name* then]**

The common IP firewall actions are as follows:

```
log;
logical-system logical-system-name <routing-instance routing-instance-name>
   <topology topology-name>;
port-mirror;
port-mirror-instance instance-name;
routing-instance routing-instance-name> <topology topology-name>;
sample;
service-filter-hit;
syslog;
topology topology-name;
```

Common IPv4 Firewall Actions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit firewall] Hierarchy" on page 55 instead of the statements being repeated.

- **[edit firewall family inet filter *filter-name* term *term-name* then]**
- **[edit firewall filter *filter-name* term *term-name* then]**

The common IP version 4 (IPv4) firewall actions are as follows:

```
(accept | discard <accounting collector-name> | reject <administratively-prohibited |
   bad-host-tos | bad-network-tos | fragmentation-needed | host-prohibited |
   host-unknown | host-unreachable | network-prohibited | network-unknown |
   network-unreachable | port-unreachable | precedence-cutoff | precedence-violation |
   protocol-unreachable | source-host-isolated | source-route-failed | tcp-reset>);
ipsec-sa sa-name;
load-balance sa-name;
next-hop-group group-name;
prefix-action action-name;
```

Common IP Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit firewall] Hierarchy" on page 55 instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in "Complete [edit firewall] Hierarchy" on page 55)
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall family inet6 dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in "Complete [edit firewall] Hierarchy" on page 55)
- **[edit firewall family inet6 filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IP firewall match conditions are as follows:

```
address {
    ip-prefix</prefix-length> <except>;
}
destination-address {
    ip-prefix</prefix-length> <except>;
}
destination-class [ class-names ] | destination-class-except [ class-names ]);
(destination-port [ port-names ] | destination-port-except [ port-names ]);
destination-prefix-list {
    list-name <except>;
}
(forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
(icmp-code [ codes ] | icmp-code-except [ codes ]);
(icmp-type [ types ] | icmp-type-except [ types ]);
interface interface-name;
(interface-group [ group-names ] | interface-group-except [ group-names ]);
interface-set set-name;
(loss-priority [ priorities ] | loss-priority-except [ priorities ]);
(packet-length [ values ] | packet-length-except [ values ]);
(port [ port-names ] | port-except [ port-names ]);
prefix-list {
    list-name <except>;
}
service-filter-hit;
source-address {
    ip-prefix</prefix-length> <except>;
}
(source-class [ class-names ] | source-class-except [ class-names ]);
(source-port [ port-names ] | source-port-except [ port-names ]);
source-prefix-list {
    list-name <except>;
}
tcp-established;
tcp-flags flag;
tcp-initial;
```


Common IPv4 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit firewall] Hierarchy" on page 55 instead of the statements being repeated.

- **[edit firewall family inet dialer-filter *filter-name* term *term-name* from]** (with the exceptions noted at this level in "Complete [edit firewall] Hierarchy" on page 55)
- **[edit firewall family inet filter *filter-name* term *term-name* from]**
- **[edit firewall filter *filter-name* term *term-name* from]**

The common IPv4 firewall match conditions are as follows:

```
(ah-spi [ values ] | ah-spi-except [ values ]);
(dscp [ code-point-values ] | dscp-except [ code-point-values ]);
(esp-spi [ values ] | esp-spi-except [ values ]);
first-fragment;
fragment-flags flag;
(fragment-offset [ offsets ] | fragment-offset-except [ offsets ]);
(ip-options [ option-names ] | ip-options-except [ option-names ]);
is-fragment;
(precedence [ precedence-names ] | precedence-except [ precedence-names ]);
(protocol [ protocol-names ] | protocol-except [ protocol-names ]);
(ttl [ ttl-values ] | ttl-except [ ttl-values ]);
```

Common Layer 2 Firewall Match Conditions

This section lists statements that are valid at the following hierarchy levels, and is referenced at those levels in "Complete [edit firewall] Hierarchy" on page 55 instead of the statements being repeated.

- **[edit firewall family ethernet-switching filter *filter-name* term *term-name* from]**
- **[edit firewall family vpls filter *filter-name* term *term-name* from]**

The common Layer 2 firewall match conditions are as follows:

```
destination-mac-address {
    mac-address <except>;
}
(destination-port [ port-names ] | destination-port-except [ port-names ]);
(dscp [ code-point-values ] | dscp-except [ code-point-values ]);
(ether-type [ protocol-types ] | ether-type-except [ protocol-types ]);
(forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
(icmp-code [ codes ] | icmp-code-except [ codes ]);
(icmp-type [ types ] | icmp-type-except [ types ]);
(interface-group [ group-names ] | interface-group-except [ group-names ]);
ip-address {
    ip-prefix</prefix-length> <except>;
}
ip-destination-address {
    ip-prefix</prefix-length> <except>;
}
```

```
(ip-precedence [ precedence-names ] | ip-precedence-except [ precedence-names ]);
(ip-protocol [ protocol-names ] | ip-protocol-except [ protocol-names ]);
ip-source-address ip-prefix</prefix-length>;
(learn-vlan-1p-priority [ priorities ] | learn-vlan-1p-priority [ priorities ]);
(learn-vlan-id [ vlan-ids ] | learn-vlan-id-except [ vlan-ids ]);
(loss-priority [ priorities ] | loss-priority-except [ priorities ]);
(port [ port-names ] | port-except [ port-names ]);
source-mac-address {
  mac-address <except>;
}
(source-port [ port-names ] | source-port-except [ port-names ]);
tcp-flags flag;
(traffic-type [ broadcast known-unicast multicast unknown-unicast ] |
  traffic-type-except [ broadcast known-unicast multicast unknown-unicast ]);
(user-vlan-1p-priority [ priorities ] | user-vlan-1p-priority [ priorities ]);
(user-vlan-id [ vlan-ids ] | user-vlan-id-except [ vlan-ids ]);
(vlan-ether-type [ protocol-types ] | vlan-ether-type-except [ protocol-types ]);
```

Complete [edit firewall] Hierarchy

```
firewall {
  family (any | ccc | ethernet-switching | inet | inet6 | mpls | vpls) {
    ... the family subhierarchies appear after the main [edit firewall] hierarchy ...
  }
  filter filter-name {
    accounting-profile [ profile-names ];
    enhanced-mode;
    interface-shared-with;
    interface-specific;
    physical-interface-policer;
    term term-name {
      filter filter-name;
      from {
        ... statements in Common IP Firewall Match Conditions on page 52 AND
          statements in Common IPv4 Firewall Match Conditions on page 53 ...
      }
      then {
        ... statements in Common Firewall Actions on page 50 AND
          statements in Common IP Firewall Actions on page 51 AND
          statements in Common IPv4 Firewall Actions on page 51 ...
      }
    }
  }
  hierarchical-policer policer-name {
    aggregate {
      if-exceeding {
        bandwidth-limit bps;
        burst-size-limit bytes;
      }
      then {
        discard;
        forwarding-class class-name;
        loss-priority (high | low | medium-high | medium-low);
      }
    }
    logical-interface-policer;
    physical-interface-policer;
    premium {
      if-exceeding {
        bandwidth-limit bps;
        burst-size-limit bytes;
      }
      then {
        discard;
      }
    }
  }
  shared-bandwidth-policer;
    interface-set interface-set-name {
    interface-name;
  }
  load-balance-group group-name {
    next-hop-group [ group-names ];
```

```
    }
    policer policer-name {
      filter-specific;
      if-exceeding {
        (bandwidth-limit bps | bandwidth-percent percentage);
        burst-size-limit bytes;
      }
      logical-bandwidth-policer;
      logical-interface-policer;
      physical-interface-policer;
      then {
        discard;
        forwarding-class class-name;
        loss-priority (high | low | medium-high | medium-low);
      }
    }
    three-color-policer policer-name {
      action {
        loss-priority high then discard;
      }
      filter-specific;
      logical-interface-policer;
      physical-interface-policer;
      shared-bandwidth-policer;
      single-rate {
        (color-aware | color-blind);
        committed-burst-size bytes;
        committed-information-rate bps;
        excess-burst-size bytes;
      }
      two-rate {
        (color-aware | color-blind);
        committed-burst-size bytes;
        committed-information-rate bps;
        peak-burst-size bytes;
        peak-information-rate bps;
      }
    }
}

firewall {
    family any {
      filter filter-name {
        interface-shared;
        term term-name {
          from {
            (forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
            interface interface-name;
            interface-set set-name;
            (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
            (packet-length [ values ] | packet-length-except [ values ]);
          }
          then {
            ... statements in Common Firewall Actions on page 50 PLUS ...
            (accept | discard);
          }
```

```
        }
      }
    }
}


firewall {
  family ccc {
    filter filter-name {
      accounting-profile [ profile-names ];
      physical-interface-filter;
      interface-specific;
      term term-name {
        filter filter-name;
        from {
          (forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
          (interface-group [ group-names ] | interface-group-except [ group-names ]);
          (learn-vlan-1p-priority [ priorities ] | learn-vlan-1p-priority [ priorities ]);
          (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
          (user-vlan-1p-priority [ priorities ] | user-vlan-1p-priority [ priorities ]);
        }
        then {
          ... statements in Common Firewall Actions on page 50 PLUS ...
          (accept | discard);
          port-mirror-instance instance-name;
        }
      }
    }
  }
}

firewall {
  family ethernet-switching {
    filter filter-name {
      interface-specific;
      term term-name {
        from {
          destination-address {
            ip-prefix</prefix-length>;
          }
          destination-mac-address {
            mac-address;
          }
          destination-port [ port-names ];
          destination-prefix-list {
            list-name;
          }
          dot1q-tag [ tag-values ];
          dot1q-user-priority [ priority-values ];
          dscp [ code-point-values ];
          ether-type [ protocol-names ];
          fragment-flags flag;
          icmp-code [ codes ];
          icmp-type [ types ];
          interface interface-name;
          is-fragment;
```

```
            precedence [ precedence-names ];
            protocol [ protocol-names ];
            source-address {
              ip-prefix</prefix-length>;
            }
            source-mac-address {
              mac-address;
            }
            source-port [ port-names ];
            source-prefix-list {
              list-name;
            }
            tcp-established;
            tcp-flags flag;
            tcp-initial;
            vlan [ vlan-names ];
          }
          then {
            (accept | discard);
            analyzer analyzer-name;
            count counter-name;
            forwarding-class class-name;
            interface interface-name;
            log;
            loss-priority (high | low);
            policer policer-name;
            syslog;
            vlan vlan-name;
          }
        }
      }
    }
  }

  firewall {
    family inet {
      dialer-filter filter-name {
        accounting-profile [ profile-names ];
        term term-name {
          from {
            ... statements in Common IP Firewall Match Conditions on page 52 AND
              statements in Common IPv4 Firewall Match Conditions on page 53 EXCEPT
              FOR ...
              (ah-spi [ values ] | ah-spi-except [ values ]);   # NOT valid at this level
              (destination-class [ class-names ] |
                destination-class-except [ class-names ]);   # NOT valid at this level
              interface interface-name;   # NOT valid at this level
              (loss-priority [ priorities ] | loss-priority-except [ priorities ]);   # NOT valid at
                this level
              service-filter-hit;   # NOT valid at this level
              (source-class [ class-names ] | source-class-except [ class-names ]);   # NOT
                valid at this level
          }
          then {
            (ignore | note);
            log;
```

```
        sample;
        syslog;
      }
    }
  }
  filter filter-name {
    accounting-profile [ profile-names ];
    interface-specific;
    term term-name {
      filter filter-name;
      from {
        ... statements in Common IP Firewall Match Conditions on page 52 AND
          statements in Common IPv4 Firewall Match Conditions on page 53 ...
      }
      then {
        ... statements in Common Firewall Actions on page 50 AND
          statements in Common IP Firewall Actions on page 51 AND
          statements in Common IPv4 Firewall Actions on page 51 ...
      }
    }
  }
  prefix-action name {
    count;
    destination-prefix-length prefix-length;
    filter-specific;
    policer policer-name;
    source-prefix-length prefix-length;
    subnet-prefix-length prefix-length;
  }
  service-filter filter-name {
    term term-name {
      from {
        address {
          ip-prefix</prefix-length>;
        }
        (ah-spi [ values ] | ah-spi-except [ values ]);
        destination-address {
          ip-prefix</prefix-length>;
        }
        (destination-port [ port-names ] | destination-port-except [ port-names ]);
        destination-prefix-list {
          list-name;
        }
        (esp-spi [ values ] | esp-spi-except [ values ]);
        first-fragment;
        fragment-flags flag;
        (fragment-offset [ offsets ] | fragment-offset-except [ offsets ]);
        (interface-group [ group-names ] | interface-group-except [ group-names ]);
        (ip-options [ option-names ] | ip-options-except [ option-names ]);
        is-fragment;
        (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
        (port [ port-names ] | port-except [ port-names ]);
        prefix-list {
          list-name;
        }
        (protocol [ protocol-names ] | protocol-except [ protocol-names ]);
```

```
            source-address {
              ip-prefix</prefix-length>;
            }
            (source-port [ port-names ] | source-port-except [ port-names ]);
            source-prefix-list {
              list-name;
            }
            tcp-flags flag-name;
          }
          then {
            count counter-name;
            log;
            port-mirror;
            sample;
            (service | skip);
          }
        }
      }
      simple-filter filter-name {
        term term-name {
          from {
            destination-address ip-prefix</prefix-length>;
            destination-port port-name;
            forwarding-class [ class-names ];
            protocol protocol-name;
            source-address ip-prefix</prefix-length>;
            source-port port-name;
          }
          then {
            forwarding-class class-name;
            loss-priority (high | low | medium-high | medium-low);
            policer policer-name;
          }
        }
      }
    }
  }

  firewall {
    family inet6 {
      dialer-filter filter-name {
        accounting-profile [ profile-names ];
        term term-name {
          from {
            ... statements in Common IP Firewall Match Conditions on page 52 PLUS ...
              (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
            ... BUT NOT ...
              (destination-class [ class-names ] |
                destination-class-except [ class-names ]);   # NOT valid at this level
              (forwarding-class [ class-names ] |
                forwarding-class-except [ class-names ]);   # NOT valid at this level
              interface interface-name;   # NOT valid at this level
              (interface-group [ group-names ] | interface-group-except [ group-names ]);  #
                NOT valid at this level
              (loss-priority [ priorities ] | loss-priority-except [ priorities ]);   # NOT valid at
                this level
```

```
            service-filter-hit;   # NOT valid at this level
            (source-class [ class-names ] | source-class-except [ class-names ]);   # NOT
              valid at this level
            tcp-established;   # NOT valid at this level
            tcp-flags flag;   # NOT valid at this level
            tcp-initial;   # NOT valid at this level
        }
        then {
          (ignore | note);
          log;
          sample;
          syslog;
        }
      }
    }
    filter filter-name {
      accounting-profile [ profile-names ];
      interface-specific;
      term term-name {
        filter filter-name;
        from {
          ... statements in Common IP Firewall Match Conditions on page 52 PLUS ...
          (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
          (traffic-class [ code-point-values ] | traffic-class-except [ code-point-values ]);
        }
        then {
          ... statements in Common Firewall Actions on page 50 AND
            statements in Common IP Firewall Actions on page 51 PLUS ...
          (accept | discard | reject <address-unreachable | administratively-prohibited |
            beyond-scope | fragmentation-needed | no-route | port-unreachable |
            tcp-reset>);
        }
      }
    }
    service-filter filter-name {
      term term-name {
        from {
          address {
            ip-prefix</prefix-length>;
          }
          (ah-spi [ values ] | ah-spi-except [ values ]);
          destination-address {
            ip-prefix</prefix-length>;
          }
          (destination-port [ port-names ] | destination-port-except [ port-names ]);
          destination-prefix-list {
            list-name;
          }
          (esp-spi [ values ] | esp-spi-except [ values ]);
          (interface-group [ group-names ] | interface-group-except [ group-names ]);
          (next-header [ protocol-types ] | next-header-except [ protocol-types ]);
          (port [ port-names ] | port-except [ port-names ]);
          prefix-list {
            list-name;
          }
          source-address {
```

```
                ip-prefix</prefix-length>;
              }
              (source-port [ port-names ] | source-port-except [ port-names ]);
              source-prefix-list {
                list-name;
              }
              tcp-flags flag-name;
            }
            then {
              count counter-name;
              log;
              port-mirror;
              sample;
              (service | skip);
            }
          }
        }
      }
    }

    firewall {
      family mpls {
        filter filter-name {
          accounting-profile [ profile-names ];
          interface-specific;
          physical-interface-filter;
          term term-name {
            from {
              (exp [ exp-bits ] | exp-except [ exp-bits ]);
            }
            then {
              (ignore | note);
              log;
              sample;
              syslog;
            }
          }
        }
        filter filter-name {
          accounting-profile [ profile-names ];
          interface-specific;
          physical-interface-filter;
          term term-name {
            filter filter-name;
            from {
              (exp [ exp-bits ] | exp-except [ exp-bits ]);
              (forwarding-class [ class-names ] | forwarding-class-except [ class-names ]);
              interface interface-name;
              interface-set set-name;
              (loss-priority [ priorities ] | loss-priority-except [ priorities ]);
            }
            then {
              ... statements in Common Firewall Actions on page 50 PLUS ...
              (accept | discard);
              sample;
            }
```

```
        }
      }
    }
  }

  firewall {
    family vpls {
      filter filter-name {
        accounting-profile [ profile-names ];
        interface-specific;
        term term-name {
          filter filter-name;
          from {
            ... statements in Common Layer 2 Firewall Match Conditions on page 53 ...
          }
          then {
            ... statements in Common Firewall Actions on page 50 PLUS ...
            (accept | discard);
            port-mirror;
            port-mirror-instance instance-name;
          }
        }
      }
    }
  }
```

Related Documentation

- Notational Conventions Used in Junos OS Configuration Hierarchies

action

Syntax

```
action {
  loss-priority high then discard;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall three-color-policer *name*],
[edit firewall three-color-policer *name*],
[edit logical-systems *logical-system-name* firewall three-color-policer *name*]

Release Information

Statement introduced in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Support at the **[edit dynamic-profiles ... three-color-policer]** hierarchy level introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Discard traffic on a logical interface using tricolor marking policing.

NOTE: This statement is supported only on IQ2 interfaces.

The remaining statement is explained separately.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Three-Color Policer Configuration Overview
- Basic Single-Rate Three-Color Policers
- Basic Two-Rate Three-Color Policers
- Two-Color and Three-Color Logical Interface Policers
- Two-Color and Three-Color Physical Interface Policers
- Two-Color and Three-Color Policers at Layer 2
- loss-priority high then discard

family (Multifield Classifier)

Syntax

```
family family-name {
  filter filter-name {
    term term-name {
      from {
        match-conditions;
      }
      then {
        dscp 0;
        forwarding-class class-name;
        loss-priority (high | low);
        three-color-policer {
          (single-rate | two-rate) policer-name;
        }
      }
    }
  }
}
```

Hierarchy Level [edit firewall]

Release Information Statement introduced before Junos OS Release 7.4.

Description Configure a firewall filter for IP version 4 (IPv4) or IP version 6 (IPv6) traffic.

Options *family-name*—Protocol family:

- **ccc**—Circuit cross-connect parameters
- **inet**—IPv4 parameters
- **inet6**—IPv6 protocol parameters
- **iso**—OSI ISO protocol parameters
- **mpls**—MPLS protocol parameters
- **tcc**—Translational cross-connect parameters
- **vpls**—Virtual private LAN service parameters.

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 Multifield Classifiers

filter (Configuring)

Syntax

```
filter filter-name {
  accounting-profile name;
  enhanced-mode;
  interface-shared;
  interface-specific;
  physical-interface-filter;
  term term-name {
    filter filter-name;
    from {
      match-conditions;
    }
    then {
      actions;
    }
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall family *family-name*],
[edit firewall family *family-name*],
[edit logical-systems *logical-system-name* firewall family *family-name*]

Release Information

Statement introduced before Junos OS Release 7.4.
Logical systems support introduced in Junos OS Release 9.3.
physical-interface-filter statement introduced in Junos OS Release 9.6.
Support at the **[edit dynamic-profiles ... family family-name]** hierarchy level introduced in Junos OS Release 11.4.
Support for the **interface-shared**> statement introduced in Junos OS Release 12.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure firewall filters.

Options

filter-name—Name that identifies the filter. This must be a non-reserved string of not more than 64 characters. To include spaces in the name, enclose it in quotation marks (" "). In Junos OS Release 9.0 and later, you can no longer use special characters within the name of a firewall filter. Firewall filter names are restricted from having the form **__.*__** (beginning and ending with underscores) or **__.*** (beginning with an underscore.

The remaining statements are explained separately.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Guidelines for Configuring Standard Firewall Filters
- Guidelines for Applying Standard Firewall Filters
- Configuring Multifield Classifiers
- Using Multifield Classifiers to Set PLP on page 28

- simple-filter

logical-interface-policer

Syntax logical-interface-policer;

Hierarchy Level
[edit dynamic-profiles *profile-name* firewall policer *policer-name*],
[edit dynamic-profiles *profile-name* firewall three-color-policer *name*],
[edit firewall atm-policer*atm-policer-name*]
[edit firewall policer *policer-name*],
[edit firewall policer *policer-template-name*],
[edit firewall three-color-policer *policer-name*],
[edit logical-systems *logical-system-name* firewall policer *policer-name*],
[edit logical-systems *logical-system-name* firewall three-color-policer *name*]

Release Information
Statement introduced before Junos OS Release 7.4.
Support at the **[edit firewall three-color-policer *policer-name*]** hierarchy level introduced in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Support at the **[edit dynamic-profiles ... policer *policer-name*] and [edit dynamic-profiles ... three-color-policer *name*]** hierarchy levels introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Configure a logical interface policer.

NOTE: Starting in Junos OS Release 12.2R2, on T Series Core Routers only, you can configure an MPLS LSP policer for a specific LSP to be shared across different protocol family types. You must include the logical-interface-policer statement to do so.

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

Related Documentation

- Two-Color and Three-Color Logical Interface Policers
- Traffic Policer Types
- Configuring Tricolor Marking Policers on page 21
- action on page 64
- Configuring Gigabit Ethernet Two-Color and Tricolor Policers
- action

loss-priority (Normal Filter)

Syntax loss-priority (high | low);

Hierarchy Level [edit firewall family *family-name* filter *filter-name* term *term-name* then]

Release Information Statement introduced before Junos OS Release 7.4.

Description Set the loss priority of incoming packets.

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

Related Documentation
- Configuring Multifield Classifiers

loss-priority (Simple Filter)

Syntax loss-priority (high | low | medium);

Hierarchy Level [edit firewall family *family-name* simple-filter *filter-name* term *term-name* then]

Release Information Statement introduced in Junos OS Release 7.6.

Description Set the loss priority of incoming packets.

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

Related Documentation
- Configuring Multifield Classifiers

policer (Configuring)

Syntax

```
policer policer-name {
  filter-specific;
  if-exceeding {
    bandwidth-limit bps;
    bandwidth-percent number;
    burst-size-limit bytes;
  }
  logical-bandwidth-policer;
  logical-interface-policer;
  physical-interface-policer;
  shared-bandwidth-policer;
  then {
    policer-action;
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall],
[edit firewall],
[edit logical-systems *logical-system-name* firewall]

Release Information

Statement introduced before Junos OS Release 7.4.
The **out-of-profile** policer action added in Junos OS Release 8.1.
The **logical-bandwidth-policer** statement added in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
The **physical-interface-policer** statement introduced in Junos OS Release 9.6.
The **shared-bandwidth-policer** statement added in Junos OS Release 11.2
Support at the **[edit dynamic-profiles ... firewall]** hierarchy level introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure policer rate limits and actions. When included at the **[edit firewall]** hierarchy level, the **policer** statement creates a template, and you do not have to configure a policer individually for every firewall filter or interface. To activate a policer, you must include the **policer-action** modifier in the **then** statement in a firewall filter term or on an interface.

Options

policer-action—One or more actions to take:

- **discard**—Discard traffic that exceeds the rate limits.
- **forwarding-class *class-name***—Specify the particular forwarding class.
- **loss-priority**—Set the packet loss priority (PLP) to **low**, **medium-low**, **medium-high**, or **high**.
- **out-of-profile**—On J Series routers with strict priority queuing, prevent starvation of other queues by rate limiting the data stream entering the strict priority queue, marking the packets that exceed the rate limit as out-of-profile, and dropping the out-of-profile packets if the physical interface is congested.

policer-name—Name that identifies the policer. The name can contain letters, numbers, and hyphens (-), and can be up to 255 characters long. To include spaces in the name, enclose it in quotation marks (" "). Policer names cannot begin with an underscore in the form **__.***.

then—Actions to take on matching packets.

The remaining statements are explained separately.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Bandwidth Policer Overview
- Configuring Multifield Classifiers
- Logical Interface (Aggregate) Policer Overview
- Physical Interface Policer Overview
- Statement Hierarchy for Configuring Policers
- Single-Rate Two-Color Policer Overview
- Using Multifield Classifiers to Set PLP on page 28
- **filter (Configuring) on page 66**
- priority (Schedulers)

shared-bandwidth-policer

Syntax shared-bandwidth-policer;

Hierarchy Level [edit firewall policer *policer-name*]

[edit firewall three-color-policer *policer-name*]
[edit firewall hierarchical-policer *policer-name*]

Release Information Statement introduced in Junos OS Release 11.2.
Support for MX Series MPC and MIC interfaces added in Junos OS Release 12.1.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Policer instances share bandwidth. This enables configuration of interface-specific policers applied on an aggregated Ethernet bundle or an aggregated SONET bundle to match the effective bandwidth and burst-size to user-configured values. This feature is supported on the following platforms: T Series routers, M120, M10i, M7i (CFEB-E only), M320 (SFPC only), MX240, MX480, and MX960 with DPC, MPC, and MIC interfaces, and EX Series switches.

NOTE: **This statement is not supported on T4000 Type 5 FPCs.**

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

Related Documentation

- Policer Support for Aggregated Ethernet Bundle Overview on page 9

term (Normal Filter)

Syntax

```
term term-name {
  from {
    match-conditions;
  }
  then {
    forwarding-class class-name;
    loss-priority (high | low);
    three-color-policer {
      (single-rate | two-rate) policer-name;
    }
  }
}
```

Hierarchy Level [edit firewall family *family-name* filter *filter-name*]

Release Information Statement introduced before Junos OS Release 7.4.

Description Define a firewall filter term.

Options

from—Match packet fields to values. If not included, all packets are considered to match and the actions and action modifiers in the **then** statement are taken.

match-conditions—One or more conditions to use to make a match.

term-name—Name that identifies the term. The name can contain letters, numbers, and hyphens (-), and can be up to 255 characters long. To include spaces in the name, enclose it in quotation marks (“ ”).

then—Actions to take on matching packets. If not included and a packet matches all the conditions in the **from** statement, the packet is accepted. For CoS, only the actions listed are allowed. These statements are explained separately.

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

Related Documentation

- Configuring Multifield Classifiers

then

Syntax

```
then {
  application-profile profile-name;
  dscp (alias | bits);
  forwarding-class class-name;
  syslog;
  (reflexive | reverse) {
    application-profile profile-name;
    dscp (alias | bits);
    forwarding-class class-name;
    syslog;
  }
}
```

Hierarchy Level [edit services cos rule *rule-name* term *term-name*]

Release Information Statement introduced in Junos OS Release 8.1.

Description Define the CoS term actions.

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 Actions in a CoS Rule
- Configuring Actions in CoS Rules

three-color-policer (Applying)

Syntax

```
three-color-policer {
    (single-rate | two-rate) policer-name;
}
```

Hierarchy Level

[edit firewall family *family-name* filter *filter-name* term *term-name* then]
[edit logical-systems *logical-system-name* firewall family *family-name* filter *filter-name* term *term-name* then]

Release Information

Statement introduced in Junos OS Release 7.4.
single-rate statement added in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

For M320 and T Series routers with Enhanced II Flexible PIC Concentrators (FPCs) and the T640 router with Enhanced Scaling FPC4, apply a tricolor marking policer.

Options

single-rate—Named tricolor policer is a single-rate policer.

two-rate—Named tricolor policer is a two-rate policer.

policer-name—Name of a tricolor policer.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Applying Tricolor Marking Policers to Firewall Filters on page 22
- Standard Firewall Filter Nonterminating Actions
- Three-Color Policer Configuration Overview

three-color-policer (Configuring)

Syntax

```
three-color-policer policer-name {
  action {
    loss-priority high then discard;
  }
  filter-specific;
  logical-interface-policer;
  physical-interface-policer;
  shared-bandwidth-policer;
  single-rate {
    (color-aware | color-blind);
    committed-burst-size bytes;
    committed-information-rate bps;
    excess-burst-size bytes;
  }
  two-rate {
    (color-aware | color-blind);
    committed-burst-size bytes;
    committed-information-rate bps;
    peak-burst-size bytes;
    peak-information-rate bps;
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* firewall],
[edit firewall],
[edit logical-systems *logical-system-name* firewall]

Release Information

Statement introduced before Junos OS Release 7.4.
The **action** and **single-rate** statements added in Junos OS Release 8.2.
Logical systems support introduced in Junos OS Release 9.3.
Support at the **[edit dynamic-profiles ... firewall]** hierarchy level introduced in Junos OS Release 11.4.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

Configure a three-color policer.

Options

policer-name—Name of the three-color policer. Reference this name when you apply the policer to an interface.

The remaining statements are explained separately.

Required Privilege Level

firewall—To view this statement in the configuration.
firewall-control—To add this statement to the configuration.

Related Documentation

- Statement Hierarchy for Configuring Policers
- Configuring Tricolor Marking Policers on page 21
- Three-Color Policer Configuration Guidelines
- Basic Single-Rate Three-Color Policers

- Basic Two-Rate Three-Color Policers
- Two-Color and Three-Color Logical Interface Policers
- Two-Color and Three-Color Physical Interface Policers
- Two-Color and Three-Color Policers at Layer 2

[edit interfaces] Hierarchy Level

The following statement hierarchy can also be included at the **[edit logical-systems *logical-system-name*]** hierarchy level.

```
interfaces {
  interface-name {
    ... the "interface-name" subhierarchy appears after the main [edit interfaces] hierarchy
      level ...
  }
  interface-set interface-set-name {
    interface interface-name {
      (unit unit-number | vlan-tags-outer vlan-tag);
    }
  }
  irb (Interfaces) {
    accounting-profile name;
    description  text;
    disable;

    (gratuitous-arp-reply | no-gratuitous-arp-reply);
    hold-time up milliseconds down milliseconds;
    mtu bytes;
    no-gratuitous-arp-request;

    traceoptions {
      flag flag;
    }
    (traps | no-traps);
    unit logical-unit-number {
      accounting-profile name;
      bandwidth rate;
      description text;
      disable;
      encapsulation type;
      family inet {
        accounting {
          destination-class-usage;
          source-class-usage {
            input;
            output;
          }
        }
        address ipv4-address {
          arp ip-address (mac | multicast-mac) mac-address <publish>;
          broadcast address;
```

```
        preferred;
        primary;
        vrrp-group group-id {
          (accept-data | no-accept-data);
          advertise-interval seconds;
          advertisements-threshold number;
          authentication-key key;
          authentication-type authentication;
          fast-interval milliseconds;
          (preempt | no-preempt) {
            hold-time seconds;
          }
          priority number;
          track {
            interface interface-name {
              bandwidth-threshold bits-per-second priority-cost priority;
              priority-cost priority;
            }
            priority-hold-time seconds;
            route prefix/prefix-length routing-instance instance-name priority-cost priority;
          }
          virtual-address [ addresses ];
          vrrp-inherit-from vrrp-group;
        }
      }
      filter {
        input filter-name;
        output filter-name;
      }
      mtu bytes;
      no-neighbor-learn;
      no-redirects;
      primary;
      rpf-check {
        fail-filter filter-name;
        mode {
          loose;
        }
      }
      targeted-broadcast {
        forward-and-send-to-re;
        forward-only;
      }
    }
    family inet6 {
      accounting {
        destination-class-usage;
        source-class-usage {
          input;
          output;
        }
      }
      address address {
        eui-64;
        ndp ip-address (mac | multicast-mac) mac-address <publish>;
        preferred;
```

```
        primary;
        vrrp-inet6-group group-id {
          accept-data | no-accept-data;
          advertisements-threshold number;
          authentication-key key;
          authentication-type authentication;
          fast-interval milliseconds;
          inet6-advertise-interval milliseconds;
          preempt | no-preempt {
            hold-time seconds;
          }
          priority number;
          track {
            interface interface-name {
              bandwidth-threshold bandwidth priority-cost number;
              priority-cost number;
            }
            priority-hold-time seconds;
            route ip-address/mask routing-instance instance-name priority-cost cost;
          }
          virtual-inet6-address [addresses];
          virtual-link-local-address ipv6–address;
          vrrp-inherit-from {
            active-group group-number;
            active-interface interface-name;
          }
        }
      }
      (dad-disable | no-dad-disable);
      filter {
        input filter-name;
        output filter-name;
      }
      mtu bytes;
      nd6-stale-time seconds;
      no-neighbor-learn;
      no-redirects;
      policer {
        input policer-name;
        output policer-name;
      }
      rpf-check {
        fail-filter filter-name;
        mode {
          loose;
        }
      }
    }
    family iso {
      address interface-address;
      mtu bytes;
    }
    family mpls {
      filter {
        input filter-name;
        output filter-name;
```

```
            }
            mtu bytes;
            policer {
              input policer-name;
              output policer-name;
            }
          }
          native-inner-vlan-id vlan-id;
          proxy-arp (restricted | unrestricted);
          (traps | no-traps);
          vlan-id-list [vlan-id's];
          vlan-id-range [vlan-id-range];
        }
      }
      traceoptions {
        file <filename> <files number> <match regular-expression> <size maximum-file-size>
          <world-readable | no-world-readable>;
        flag flag <disable>;
        no-remote-trace;
      }
    }

    interfaces {
      interface-name {
        disable;
        accounting-profile name;
        aggregated-ether-options {
          ethernet-switch-profile {
            tag-protocol-id [ hexadecimal-identifiers ];
          }
          (flow-control | no-flow-control);
          lacp {
            (active | passive);
            admin-key key;
            fast-failover;
            link-protection {
              disable;
              (revertive | non-revertive);
            }
            periodic (fast | slow);
            system-id mac-address;
            system-priority priority;
          }
          (link-protection | no-link-protection);
          link-speed (100m | 1g | 8g | 10g | 40g | 50g | 80g | 100g | oc192);
          logical-interface-fpc-redundancy;
          (loopback | no-loopback);
          mc-ae {
            chassis-id chassis-id;
            events {
              iccp-peer-down {
                force-icl-down;
                prefer-status-control-active;
              }
            }
            mc-ae-id mc-ae-id;
```

```
    mode (active-active | active-standby);
    redundancy-group group-id;
    status-control (active | standby);
  }
  minimum-links number;
  rebalance-periodic {
    start-time time;
    interval number;
  }
  source-address-filter {
    mac-address;
  }
  (source-filtering | no-source-filtering);
}
auto-configure {
  remove-when-no-subscribers;
  stacked-vlan-ranges {
    access-profile profile-name;
    authentication {
      password password-string;
      username-include {
        circuit-type;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        mac-address;
        option-82 ( circuit-id | remote-id);
        radius-realm radius-realm-string;
        user-prefix user-prefix-string;
      }
    }
    dynamic-profile profile-name {
      accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
      ranges (any | low-tag–high-tag),(any | low-tag–high-tag);
    }
  }
  vlan-ranges {
    access-profile profile-name;
    authentication {
      password password-string;
      username-include {
        circuit-type;
        delimiter delimiter-character;
        domain-name domain-name-string;
        interface-name;
        mac-address;
        option-82;
        radius-realm radius-realm-string;
        user-prefix user-prefix-string;
      }
    }
    dynamic-profile profile-name {
      accept (any | dhcp-v4 | dhcp-v6 | inet | inet6);
      ranges (any | low-tag)–(any | high-tag);
    }
  }
```

```
    override tag vlan-tag dynamic-profile profile name;
}
encapsulation (ethernet-bridge | ethernet-vpls | extended-vlan-bridge |
    extended-vlan-vpls | flexible-ethernet-services | vlan-vpls);
ether-options {
    802.3ad {
        aex;
        (backup | primary);
        lacp {
            force-up;
            port-priority
        }
    }
    asynchronous-notification;
    (auto-negotiation | no-auto-negotiation);
    ethernet-switch-profile {
        ethernet-policer-profile {
            input-priority-map {
                ieee802.1p premium [ values ];
            }
            output-priority-map {
                classifier {
                    premium {
                        forwarding-class class-name {
                            loss-priority (high | low);
                        }
                    }
                }
            }
            policer cos-policer-name {
                aggregate {
                    bandwidth-limit bps;
                    burst-size-limit bytes;
                }
                premium {
                    bandwidth-limit bps;
                    burst-size-limit bytes;
                }
            }
            tag-protocol-id;
        }
        (mac-learn-enable | no-mac-learn-enable);
    }
    (flow-control | no-flow-control);
    ignore-l3-incompletes;
    link-mode (automatic | full-duplex | half-duplex);
    (lloopback | no-loopback);
    keepalives <interval seconds> <down-count number> <up-count number>;
    speed (1g | 10m | 100m | 10m-100m | auto-negotiation);
    source-address-filter {
        mac-address;
    }
    source-filtering | no-source-filtering;
}
flexible-vlan-tagging;
(gratuitous-arp-reply | no-gratuitous-arp-reply);
```

```
    hold-time (up milliseconds | down milliseconds);
    interface-transmit-statistics;
    (keepalives <down-count number> <interval seconds> <up-count number> |
      no-keepalives);
    layer2-policer {
      apply-groups [ group-names ];
      apply-groups-except [ group-names ];
    }
    link-mode (automatic | full-duplex);
    mac mac-address;
    mtu bytes;
    multi-chassis-protection peer-ip-address {
      interface interface-name;
    }
    native-vlan-id number;
    no-gratuitous-arp-request;
    optics-options {
      alarm low–light–alarm {
        (link-down | syslog);
      }
      warning low–light–warning {
        (link-down | syslog);
      }
      wavelength nm;
    }
    passive-monitor-mode;
    per-unit-scheduler;
    speed (10m | 100m | 1g | auto | oc3 | oc12 | oc48);
    stacked-vlan-tagging;
    traceoptions {
      flag flag;
    }
    transmit-bucket {
      overflow discard;
      rate percentage;
      threshold bytes;
    }
    (traps | no-traps);
    unidirectional;
    vlan-tagging;
  }


  interface-name {
    unit logical-unit-number {
      disable;
      accept-source-mac {
        mac-address mac-address {
          policer {
            input policer-name;
            output policer-name;
          }
        }
      }
      account-layer2-overhead (Interface Level) {
        value;
```

```
    egress bytes;
    ingress bytes;
}
accounting-profile name;
advisory-options {
    downstream-rate rate;
    upstream-rate rate;
}
arp-resp (restricted|unrestricted);
bandwidth rate;
clear-dont-fragment-bit;
copy-tos-to-outer-ip-header;
demux-destination family;
encapsulation (vlan-bridge | vlan-vpls);
epd-threshold cells plp1 cells;
filter filter-name;
inner-vlan-id-range start start-id end end-id;
input-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
interface-shared-with psdnumerical-index;
layer2-policer {
    input-hierarchical-policer policer-name;
    input-policer policer-name;
    input-three-color policer-name;
    output-policer policer-name;
    output-three-color policer-name;
}
multi-chassis-protection peer-ip-address {
    interface interface-name;
}
native-inner-vlan-id number;
output-vlan-map {
    (pop | pop-pop | pop-swap | push | push-push | swap | swap-push | swap-swap);
    inner-tag-protocol-id tpid;
    inner-vlan-id number;
    tag-protocol-id tpid;
    vlan-id number;
}
peer-interface interface-name;
peer-unit unit-number;
plp-to-clp;
proxy-arp <restricted | unrestricted>;
rpm {
    (client | server);
    twamp-server;
}
swap-by-poppush;
vlan-id number;
vlan-id-list [ vlan-id vlan-id-vlan-id ];
vlan-id-range number-number;
```

```
    vlan-tags (inner <tpid.>vlan-id | inner-list [vlan-id vlan-id–vlan-id ] |
      inner-range <tpid.>vlan-id–vlan-id) outer <tpid.>vlan-id;
}

unit logical-unit-number {
    family ethernet-switching {
        filter{
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
        (output filter-name | output-list [ filter-names ]);
        (inner-vlan-id-list [ vlan-ids ] | vlan-id number | vlan-id-list [ number
            number–number ]);
        interface-mode (access | trunk);
        policer {
            input policer-name;
            output policer-name;
        }
        vlan-rewrite {
            translate old-vlan-id new-vlan-id;
        }
        vlan {
            members [ all vlan-identifiers ];
        }
    }
    family inet {
        filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
        }
        input-hierarchical-policer policer-name;
        mac-validate (loose | strict);
        mtu bytes;
        no-neighbor-learn;
        no-redirects;
        policer {
            arp policer-template-name;
            input policer-name;
            output policer-name;
        }
        primary;
        receive-options-packets;
        receive-ttl-exceeded;
        rpf-check {
            fail-filter filter-name;
            mode loose;
        }
        sampling {
            (input | output | input output);
        }
        simple-filter {
            input filter-name;
        }
        targeted-broadcast {
            forward-and-send-to-re;
            forward-only;
```

```
  }
  unnumbered-address interface-name <destination address>
    <destination-profile profile-name> <preferred-source-address address>;

}

family inet6 {
  address ipv6-address {
    destination destination-address;
    eui-64;
    ndp ipv6-address <l2-interface interface-name> <(mac mac-address |
      multicast-mac multicast-mac-address) <publish>>;
    preferred;
    primary;
    vrrp-inet6-group group-number {
      (accept-data | no-accept-data);
      fast-interval milliseconds;
      inet6-advertise-interval seconds;
      (no-preempt; | ... the following preempt statement ...)
      preempt {
        hold-time seconds;
      }
      priority number;
      track {
        interface interface-name {
          bandwidth-threshold bits-per-second priority-cost priority;
          priority-cost priority;
        }
        priority-hold-time seconds;
        route ip-address-prefix/prefix-length routing-instance instance-name
          priority-cost priority;
      }
      virtual-inet6-address [ addresses ];
      virtual-link-local-address ipv6-address;
      vrrp-inherit-from {
        active-group group-number;
        active-interface interface-name;
      }
    }
  }
  (dad-disable | no-dad-disable);
  filter {
    group filter-group-number;
    (input filter-name | input-list [ filter-names ]);
    (output filter-name | output-list [ filter-names ]);
  }
  input-hierarchical-policer policer-name;
  mtu bytes;
  nd6-stale-time seconds;
  no-neighbor-learn;
  policer {
    input policer-name;
    output policer-name;
  }
  rpf-check {
    fail-filter filter-name;
```

```
            mode loose;
          }
          sampling {
            (input | output | input output);
          }
          unnumbered-address interface-name preferred-source-address address;

        }

        family iso {
          address iso-address;
          mtu bytes;
        }

        family mlfr-end-to-end {
          bundle logical-interface-name;
        }



        family mpls {
          filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
          }
          input-hierarchical-policer policer-name;
          maximum-labels maximum-labels;
          mtu bytes;
          policer {
            input policer-name;
            output policer-name;
          }
        }



        family vpls {
          core-facing;
          filter {
            group filter-group-number;
            (input filter-name | input-list [ filter-names ]);
            (output filter-name | output-list [ filter-names ]);
          }
          policer {
            input policer-name;
            output policer-name;
          }
        }

      }
    }
  }
```

Related Documentation
- Notational Conventions Used in Junos OS Configuration Hierarchies

filter (Applying to an Interface)

Syntax

```
filter {
  input filter-name;
  output filter-name;
}
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number* family *family*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* family *family*]

Release Information

Statement introduced before Junos OS Release 7.4.

Description

Apply a filter to an interface. You can also use filters for encrypted traffic. When you configure filters, you can configure the family **inet**, **inet6**, **mpls**, or **vpls** only.

Options

input *filter-name*—Name of one filter to evaluate when packets are received on the interface.

output *filter-name*—Name of one filter to evaluate when packets are transmitted on the interface.

Required Privilege Level

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

Related Documentation
- simple-filter
- Applying Firewall Filter Tricolor Marking Policers to Interfaces
- Example: Classifying Packets Based on Their Destination Address
- Example: Configuring and Verifying a Complex Multifield Filter
- Example: Writing Different DSCP and EXP Values in MPLS-Tagged IP Packets
- Example: Configuring a Simple Filter
- Example: Configuring a Logical Bandwidth Policer
- Example: Two-Color Policers and Shaping Rate Changes

input-policer

Syntax input-policer *policer-name*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* layer2-policer]
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* layer2-policer]

Release Information Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Apply a single-rate two-color policer to the Layer 2 input traffic at the logical interface. The **input-policer** and **input-three-color** statements are mutually exclusive.

Options *policer-name*—Name of the single-rate two-color policer that you define at the **[edit firewall]** hierarchy level.

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

Related Documentation

- Two-Color and Three-Color Policers at Layer 2
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 25
- Configuring a Gigabit Ethernet Policer
- input-three-color on page 89
- layer2-policer on page 90
- logical-interface-policer on page 67
- output-policer on page 91
- output-three-color on page 92

input-three-color

Syntax input-three-color *policer-name*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* layer2-policer]
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* layer2-policer]

Release Information Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Apply a single-rate or two-rate three-color policer to the Layer 2 input traffic at the logical interface. The **input-three-color** and **input-policer** statements are mutually exclusive.

Options ***policer-name***—Name of the single-rate or two-rate three-color policer.

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

Related Documentation

- Two-Color and Three-Color Policers at Layer 2
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 25
- Configuring a Gigabit Ethernet Policer
- input-policer on page 88
- layer2-policer on page 90
- logical-interface-policer on page 67
- output-policer on page 91
- output-three-color on page 92

layer2-policer

Syntax

```
layer2-policer {
  input-policer policer-name;
  input-three-color policer-name;
  output-policer policer-name;
  output-three-color policer-name;
}
```

Hierarchy Level

[edit interfaces *interface-name* unit *logical-unit-number*],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number*],

Release Information

Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description

For 1-Gigabit Ethernet and 10-Gigabit Ethernet IQ2 and IQ2-E interfaces on M Series, MX Series, and T Series routers, and for aggregated Ethernet, Gigabit Ethernet, and 10-Gigabit Ethernet interfaces on EX Series switches, apply Layer 2 logical interface policers. The following policers are supported:

- Two-color
- Single-rate tricolor marking (srTCM)
- Two-rate tricolor marking (trTCM)

Two-color and tricolor policers are configured at the **[edit firewall]** hierarchy level.

Options

input-policer ***policer-name***—Two-color input policer to associate with the interface. This statement is mutually exclusive with the **input-three-color** statement.

input-three-color ***policer-name***—Tricolor input policer to associate with the interface. This statement is mutually exclusive with the **input-policer** statement.

output-policer ***policer-name***—Two-color output policer to associate with the interface. This statement is mutually exclusive with the **output-three-color** statement.

output-three-color ***policer-name***—Tricolor output policer to associate with the interface. This statement is mutually exclusive with the **output-policer** statement.

Required Privilege Level

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

Related Documentation

- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 25
- Configuring Gigabit Ethernet Two-Color and Tricolor Policers

output-policer

Syntax output-policer *policer-name*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* layer2-policer],
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* layer2-policer]

Release Information Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Apply a single-rate two-color policer to the Layer 2 output traffic at the logical interface. The **output-policer** and **output-three-color** statements are mutually exclusive.

Options ***policer-name***—Name of the single-rate two-color policer that you define at the **[edit firewall]** hierarchy level.

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

Related Documentation

- Two-Color and Three-Color Policers at Layer 2
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 25
- Configuring a Gigabit Ethernet Policer
- input-policer on page 88
- input-three-color on page 89
- layer2-policer on page 90
- logical-interface-policer on page 67
- output-three-color on page 92

output-three-color

Syntax output-three-color *policer-name*;

Hierarchy Level [edit interfaces *interface-name* unit *logical-unit-number* layer2-policer]
[edit logical-systems *logical-system-name* interfaces *interface-name* unit *logical-unit-number* layer2-policer]

Release Information Statement introduced in Junos OS Release 8.2.
Statement introduced in Junos OS Release 12.3R2 for EX Series switches.

Description Apply a single-rate or two-rate three-color policer to the Layer 2 output traffic at the logical interface. The **output-three-color** and **output-policer** statements are mutually exclusive.

Options ***policer-name***—Name of the single-rate or two-rate three-color policer.

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

Related Documentation

- Two-Color and Three-Color Policers at Layer 2
- Applying Layer 2 Policers to Gigabit Ethernet Interfaces on page 25
- Configuring a Gigabit Ethernet Policer
- input-three-color on page 89
- input-policer on page 88
- layer2-policer on page 90
- logical-interface-policer on page 67
- output-policer on page 91