JUNIPER NETWORKS

Junos® OS

CoS Congestion Avoidance with RED Drop Profiles

Release
13.1

Published: 2013-02-13

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

Junos® OS CoS Congestion Avoidance with RED Drop Profiles
13.1

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

YEAR 2000 NOTICE

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

END USER LICENSE AGREEMENT

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

Table of Contents

List of Figures

List of Tables

About the Documentation

Documentation and Release Notes

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

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

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

Supported Platforms

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

- T Series
- M Series
- MX Series
- PTX Series

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

Table 1 on page xi 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 xi 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.	
J-Web GUI Conventions		
Bold text like this	Represents J-Web 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 J-Web 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

RED Drop Profiles

RED Drop Profiles Overview

You can configure two parameters to control congestion at the output stage. The first parameter defines the delay-buffer bandwidth, which provides packet buffer space to absorb burst traffic up to the specified duration of delay. Once the specified delay buffer becomes full, packets with 100 percent drop probability are dropped from the head of the buffer. For more information, see Configuring the Scheduler Buffer Size.

The second parameter defines the drop probabilities across the range of delay-buffer occupancy, supporting the random early detection (RED) process. When the number of packets queued is greater than the ability of the router to empty a queue, the queue requires a method for determining which packets to drop from the network. To address this, the Junos OS provides the option of enabling RED on individual queues.

Depending on the drop probabilities, RED might drop many packets long before the buffer becomes full, or it might drop only a few packets even if the buffer is almost full.

A *drop profile* is a mechanism of RED that defines parameters that allow packets to be dropped from the network. Drop profiles define the meanings of the loss priorities.

When you configure drop profiles, there are two important values: the queue fullness and the drop probability. The *queue fullness* represents a percentage of the memory used to store packets in relation to the total amount that has been allocated for that specific queue. Similarly, the *drop probability* is a percentage value that correlates to the likelihood that an individual packet is dropped from the network. These two variables are combined in a graph-like format, as shown in Figure 1 on page 4.

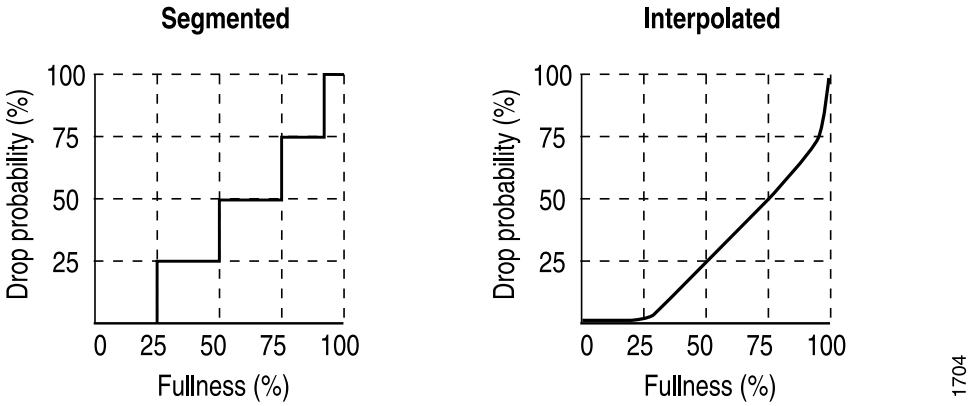
The maximum number of queue fullness levels supported per drop profile is based on the underlying line card:

- Physical or logical interfaces hosted on MICs in Queuing or Enhanced Queuing MPCs for MX Series routers support up to 64 (fill level, drop probability) pairs per segmented or interpolated drop profile.

- Physical or logical interfaces hosted on Enhanced Queuing DPCs for MX Series routers support up to 64 (fill level, drop probability) pairs per segmented drop profile or 2 pairs per interpolated drop profile. For more information, see Configuring WRED on Enhanced Queuing DPCs.
- Physical or logical interfaces hosted on IQ2 PICs or IQE PICs support up to two (fill level, drop probability) pairs per segmented or interpolated drop profile.

Figure 1 on page 4 shows both a segmented and an interpolated graph. Although the formation of these graph lines is different, the application of the profile is the same. When a packet reaches the head of the queue, a random number between 0 and 100 is calculated by the router. This random number is plotted against the drop profile using the current queue fullness of that particular queue. When the random number falls above the graph line, the packet is transmitted onto the physical media. When the number falls below graph the line, the packet is dropped from the network.

Figure 1: Segmented and Interpolated Drop Profiles



By defining multiple fill levels and drop probabilities, you create a segmented drop profile. The line segments are defined in terms of the following graphical model: in the first quadrant, the x axis represents the fill level, and the y axis represents the drop probability. The initial line segment spans from the origin (0,0) to the point (<l1>, <p1>); a second line runs from (<l1>, <p1>) to (<l2>, <p2>) and so forth, until a final line segment connects (100, 100). The software automatically constructs a drop profile containing 64 fill levels at drop probabilities that approximate the calculated line segments.

NOTE: If you configure the interpolate statement, you can specify more than 64 pairs, but the system generates only 64 discrete entries.

You specify drop probabilities in the drop profile section of the class-of-service (CoS) configuration hierarchy and reference them in each scheduler configuration. For each scheduler, you can configure multiple separate drop profiles, one for each combination of loss priority (low, medium-low, medium-high, or high) and protocol.

You can configure a maximum of 32 different drop profiles.

To configure RED drop profiles, include the following statements at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
drop-profiles {
  profile-name {
    fill-level percentage drop-probability percentage;
    interpolate {
      drop-probability [ values ];
      fill-level [ values ];
    }
  }
}
```

Default Drop Profile

By default, if you configure no drop profiles, RED is still in effect and functions as the primary mechanism for managing congestion. In the default RED drop profile, when the fill-level is 0 percent, the drop probability is 0 percent. When the fill-level is 100 percent, the drop probability is 100 percent.

As a backup method for managing congestion, tail dropping takes effect when congestion of small packets occurs. On Juniper Networks M320 Multiservice Edge Routers and T Series Core Routers, the software supports *tail-RED*, which means that when tail dropping occurs, the software uses RED to execute intelligent tail drops. On other routers, the software executes tail drops unconditionally.

Packet Loss Priority Configuration Overview

Loss priority settings help determine which packets are dropped from the network during periods of congestion. The software supports multiple packet loss priority (PLP) designations: **low** and **high**. (In addition, **medium-low** and **medium-high** PLPs are supported when you configure tricolor marking, as discussed in Configuring Tricolor Marking.) You can set PLP by configuring a behavior aggregate or multifield classifier, as discussed in Setting Packet Loss Priority and Configuring Multifield Classifiers.

NOTE: On T Series routers with different Packet Forwarding Engines (non-Enhanced Scaling and Enhanced Scaling FPCs), you can configure PLP bit copying for ingress and egress unicast and multicast traffic. To configure, include the copy-plp-all statement at the [edit class-of-service] hierarchy level.

A drop-profile map examines the loss priority setting of an outgoing packet: **high**, **medium-high**, **medium-low**, **low**, or any.

Obviously, *low*, *medium-low*, *medium-high*, and *high* are relative terms, which by themselves have no meaning. Drop profiles define the meanings of the loss priorities. In the following example, the **low-drop** drop profile defines the meaning of **low** PLP as a 10 percent drop probability when the fill level is 75 percent and a 40 percent drop probability when the fill level is 95 percent. The **high-drop** drop profile defines the meaning

of **high** PLP as a 50 percent drop probability when the fill level is 25 percent and a 90 percent drop probability when the fill level is 50 percent.

In this example, the scheduler includes two drop-profile maps, which specify that packets are evaluated by the **low-drop** drop profile if they have a **low** loss priority and are from any protocol. Packets are evaluated by the **high-drop** drop profile if they have a **high** loss priority and are from any protocol.

```
[edit class-of-service]
drop-profiles {
   low-drop {
      interpolate {
         drop-probability [ 10 40];
         fill-level [ 75 95];
      }
   }
   high-drop {
      interpolate {
         drop-probability [ 50 90];
         fill-level [ 25 50];
      }
   }
}
schedulers {
   best-effort {
      drop-profile-map loss-priority low protocol any drop-profile low-drop;
      drop-profile-map loss-priority high protocol any drop-profile high-drop;
   }
}
```

Related Documentation

- Configuring Schedulers
- Setting Packet Loss Priority
- copy-plp-all
- Configuring Multifield Classifiers

PART 2

Configuration

CHAPTER 2

Configuration Tasks

Configuring RED Drop Profiles

You enable RED by applying a drop profile to a scheduler. When RED is operational on an interface, the queue no longer drops packets from the tail of the queue. Rather, packets are dropped after they reach the head of the queue.

To configure a drop profile, include the **drop-profiles** statement at the **[edit class-of-service]** hierarchy level:

```
[edit class-of-service]
drop-profiles {
  profile-name {
    fill-level percentage drop-probability percentage;
    interpolate {
      drop-probability [ values ];
      fill-level [ values ];
    }
  }
}
```

In this configuration, include either the **interpolate** statement and its options, or the fill-level and drop-probability ***percentage*** values. These two alternatives enable you to configure either each drop probability at up to 64 fill-level/drop-probability paired values, or a profile represented as a series of line segments, as discussed in "RED Drop Profiles Overview" on page 3.

After you configure a drop profile, you must assign the drop profile to a drop-profile map, and assign the drop-profile map to a scheduler, as discussed in Configuring Drop Profile Maps for Schedulers.

Configuring Weighted RED Buffer Occupancy

By default, RED is performed based on instantaneous buffer occupancy information. However, IQ-PICs can be configured to use *weighted average* buffer occupancy information. This option is configured on a per-PIC basis and applies to the following IQ-PICs:

- Channelized T1/T3
- Channelized E1/E3
- Channelized OC3/STM1
- Channelized OC12

If you configure this feature on an unsupported PIC, you see an error message.

If you configure this feature on a channelized OC12 intelligent queuing (IQ) interface, the PIC reboots.

When weighted average buffer occupancy is configured, you configure a weight value for averaged buffer occupancy calculations. This weight value is expressed as a negative exponential value of 2 in a fractional expression. For example, a configured weight value of 2 would be expressed as $1/(2^2) = 1/4$. If a configured weight value was configured as 1 (the default), the value would be expressed as $1/(2^1) = 1/2$.

This calculated weight value is applied to the instantaneous buffer occupancy value to determine the new value of the weighted average buffer occupancy. The formula to derive the new weighted average buffer occupancy is:

new average buffer occupancy = weight value * instantaneous buffer occupancy + (1 – weight value) * current average buffer occupancy

For example, if the weight exponent value is configured as 3 (giving a weight value of $1/2^3 = 1/8$), the formula used to determine the new average buffer occupancy based on the instant buffer usage is:

new average buffer occupancy = 1/8 * instantaneous buffer occupancy + (7/8) * current average buffer occupancy

The valid operational range for the weight value on IQ-PICs is 0 through 31. A value of 0 results in the average buffer occupancy being the same as the instantaneous buffer occupancy calculations. Values higher than 31 can be configured, but in these cases the current maximum *operational* value of 31 is used for buffer occupancy calculations.

NOTE: The **show interfaces** command with the **extensive** option displays the *configured* value for the **RED buffer occupancy** weight exponent. However, in all such cases, the current *operational* maximum value of 31 is used internally.

To configure a Q-PIC for RED weighted average buffer occupancy calculations, include the **red-buffer-occupancy** statement with the **weighted-averaged** option at the **[edit chassis fpc *slot-number* pic *pic-number*]** hierarchy level:

```
[edit chassis]
fpc slot-number {
  pic pic-number {
    red-buffer-occupancy {
      weighted-averaged [ instant-usage-weight-exponent exponent-number ];
    }
```

```
    }
  }
```

Related Documentation

- Example: Configuring Weighted RED Buffer Occupancy on page 14
- red-buffer-occupancy on page 26

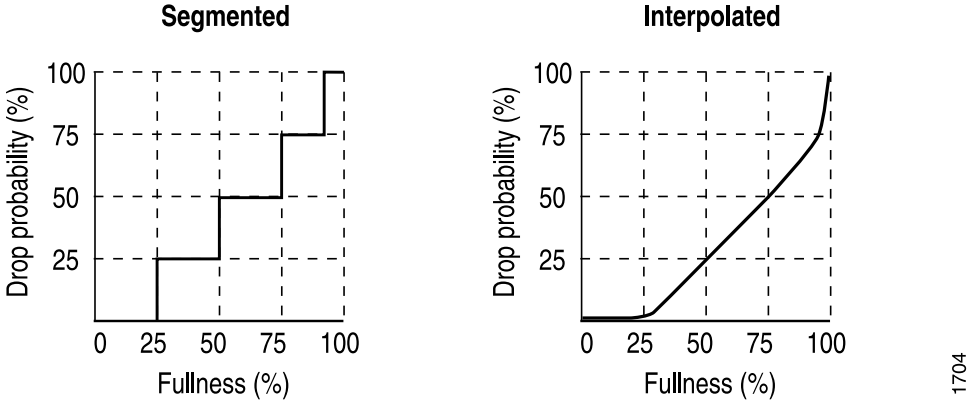
CHAPTER 3

Examples

Example: Configuring RED Drop Profiles

Create a segmented configuration and an interpolated configuration that correspond to the graphs in Figure 2 on page 13. The values defined in the configuration are matched to represent the data points in the graph line. In this example, the drop probability is 25 percent when the queue is 50 percent full. The drop probability increases to 50 percent when the queue is 75 percent full.

Figure 2: Segmented and Interpolated Drop Profiles



Creating a Segmented Configuration

```
class-of-service {
  drop-profiles {
    segmented-style-profile {
      fill-level 25 drop-probability 25;
      fill-level 50 drop-probability 50;
      fill-level 75 drop-probability 75;
      fill-level 95 drop-probability 100;
    }
  }
}
```

To create the profile's graph line, the software begins at the bottom-left corner, representing a 0 percent fill level and a 0 percent drop probability. This configuration

draws a line directly to the right until it reaches the first defined fill level, 25 percent for this configuration. The software then continues the line vertically until the first drop probability is reached. This process is repeated for all of the defined levels and probabilities until the top-right corner of the graph is reached.

Create a smoother graph line by configuring the profile with the **interpolate** statement. This allows the software to automatically generate 64 data points on the graph beginning at (0, 0) and ending at (100, 100). Along the way, the graph line intersects specific data points, which you define as follows:

Creating an Interpolated Configuration

```
class-of-service {
  drop-profiles {
    interpolated-style-profile {
      interpolate {
        fill-level [ 50 75 ];
        drop-probability [ 25 50 ];
      }
    }
  }
}
```

Example: Configuring Weighted RED Buffer Occupancy

Configure the Q-PIC to use a weight value of 1/2 in average buffer occupancy calculations.

```
[edit chassis]
fpc 0 {
  pic 1 {
    red-buffer-occupancy {
      weighted-averaged instant-usage-weight-exponent 1;
    }
  }
}
```

or

```
[edit chassis]
fpc 0 {
  pic 1 {
    red-buffer-occupancy {
      weighted-averaged; # the default value is 1 if not specified
    }
  }
}
```

Configure the Q-PIC to use a weight value of 1/4 in average buffer occupancy calculations.

```
[edit chassis]
fpc 0 {
  pic 1 {
    red-buffer-occupancy {
      weighted-averaged instant-usage-weight-exponent 2;
    }
  }
}
```

Related Documentation

- Configuring Weighted RED Buffer Occupancy on page 9
- red-buffer-occupancy on page 26

CHAPTER 4

Configuration Statements

[edit class-of-service] Hierarchy Level

This topic shows the complete configuration for class of service (CoS) statements for the **[edit class-of-service]** hierarchy level, listing all possible configuration statements and showing their level in the configuration hierarchy. When you are configuring Junos OS, your current hierarchy level is shown in the banner on the line preceding the **user@host#** prompt.

For a complete list of the Junos OS configuration statements, see the *Junos OS Hierarchy and RFC Reference*.

```
[edit class-of-service]
adjustment-control-profiles {
   profile-name {
      application {
         ancp;
         radius-coa;
         pppoe-tags;
         }
      }
   }
classifiers {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) classifier-name {
      import (classifier-name | default);
      forwarding-class class-name {
         loss-priority level code-points [ aliases ] [ bit-patterns ];
      }
   }
}
code-point-aliases {
   (dscp | dscp-ipv6 | exp | ieee-802.1 | inet-precedence) {
      alias-name bits;
   }
}
copy-plp-all;
drop-profiles {
   profile-name {
      fill-level percentage drop-probability percentage;
      interpolate {
```

```
        drop-probability [ values ];
        fill-level [ values ];
      }
    }
  }
  fabric {
    scheduler-map {
      priority (high | low) scheduler scheduler-name;
    }
  }
  forwarding-classes {
    class class-name queue-num queue-number priority (high | low);
    queue queue-number class-name priority (high | low) [ policing-priority (premium |
      normal) ];
  }
  forwarding-class-map  forwarding-class-map-name {
    class class-name queue-num queue-number [ restricted-queue queue-number ];
  }
  forwarding-policy {
    next-hop-map map-name {
      forwarding-class class-name {
        next-hop [ next-hop-name ];
        lsp-next-hop [ lsp-regular-expression ];
        non-lsp-next-hop;
        discard;
      }
    }
    class class-name {
      classification-override {
        forwarding-class class-name;
      }
    }
  }
  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;
    }
  }
  interfaces {
    interface-name {
      classifiers {
        dscp (classifier-name | default);
        ieee-802.1 (classifier-name | default)  vlan-tag (inner | outer | classifier-name);
```

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

```
            exp (rewrite-name | default)protocol protocol-types;
            exp-push-push-push default;
            exp-swap-push-push default;
            ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
            inet-precedence (rewrite-name | default)protocol protocol-types;
          }
          scheduler-map map-name;
          shaping-rate rate;
          translation-table (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 |
            to-exp-from-exp | to-inet-precedence-from-inet-precedence) table-name;
        }
      }
    }
    loss-priority-maps {
      frame-relay-de (Defining Loss Priority Maps)name {
      loss-priority levelcode-points [alias | bits ];
      }
    }
    loss-priority-rewrites {
      frame-relay-de (Defining Loss Priority Maps)name {
      loss-priority levelcode-point (alias | bits );
      }
    }
    restricted-queues {
      forwarding-class class-name queue queue-number;
    }
    rewrite-rules {
      (dscp | dscp-ipv6 | exp | ieee-802.1 | ieee-802.1ad | inet-precedence) rewrite-name {
        import (rewrite-name | default);
        forwarding-class class-name {
          loss-priority level code-point (alias | bits);
        }
      }
    }
    routing-instances routing-instance-name {
      classifiers {
        exp (classifier-name | default);
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
      }
    }
    scheduler-maps {
      map-name {
        forwarding-class class-name scheduler scheduler-name;
      }
    }
    schedulers {
      scheduler-name {
        buffer-size (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 | high);
        excess-rate percent percentage;
        excess-rate (percent percentage | proportion value);
        priority priority-level;
        transmit-rate (rate | percent percentage | remainder) <exact | rate-limit>;
```



```
    }
  }
  system-defaults {
    classifiers (classifier-name | exp)
  traffic-control-profiles profile-name {
    delay-buffer-rate (percent percentage | rate);
    excess-rate (percent percentage | proportion value);
    guaranteed-rate (percent percentage | rate);
    overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
    scheduler-map map-name;
    shaping-rate (percent percentage | rate);
  }
  translation-table {
    (to-dscp-from-dscp | to-dscp-ipv6-from-dscp-ipv6 | to-exp-from-exp |
      to-inet-precedence-from-inet-precedence) table-name {
      to-code-point value from-code-points (* | [ values ]);
    }
  }
  tri-color;
```

On Juniper Networks MX Series 3D Universal Edge Routers with Enhanced Queuing DPCs, you can configure the following CoS statements at the **[edit class-of-service interfaces]** hierarchy level:

```
interface-set interface-set-name {
  excess-bandwidth-share (proportional value | equal);
  internal-node;
  traffic-control-profiles profile-name;
  output-traffic-control-profile-remaining profile-name;
}
```

drop-probability (Interpolated Value)

Syntax drop-probability [*values*];

Hierarchy Level [edit class-of-service drop-profiles *profile-name* interpolate]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description Define up to 64 values for interpolating drop probabilities. On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.

Options ***percentage***—The probability (expressed in percentage) for a packet to be dropped from the queue.

Range: 0 through 100

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

Related Documentation

- Default Drop Profile on page 5
- Configuring CoS Tail Drop Profiles (CLI Procedure)

drop-profiles (Class-of-Service)

Syntax

```
drop-profiles {
  profile-name {
    fill-level percentage drop-probability percentage;
    interpolate {
      drop-probability [values];
      fill-level [values]
    }
  }
}
```

Hierarchy Level [edit class-of-service]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description Define drop profiles for RED.

For a packet to be dropped, it must match the drop profile. 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 packet.

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

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 RED Drop Profiles on page 9
- Understanding CoS Tail Drop Profiles
- Example: Configuring CoS on EX Series Switches
- Configuring CoS Tail Drop Profiles (CLI Procedure)

fill-level (Interpolated Value)

Syntax fill-level [*values*];

Hierarchy Level [edit class-of-service drop-profiles *profile-name* interpolate]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description Define up to 64 values for interpolating queue fill level.

On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.

Options ***values***—Data points for mapping queue fill percentage.

Range: 0 through 100

Default: In the default tail drop profile, when the fill level is 0 percent, the drop probability is 0 percent. When the fill level is 100 percent, the drop probability is 100 percent.

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

Related Documentation

- RED Drop Profiles Overview on page 3
- Configuring RED Drop Profiles on page 9.
- Understanding CoS Tail Drop Profiles
- Configuring CoS Tail Drop Profiles (CLI Procedure)

fill-level (Drop Profiles)

Syntax fill-level *percentage*;

Hierarchy Level [edit class-of-service drop-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description When configuring RED, map the fullness of a queue to a drop probability.

Options *percentage*—How full the queue is, expressed as a percentage. You configure the **fill-level** and **drop-probability** statements in pairs. To specify multiple fill levels, include multiple **fill-level** and **drop-probability** statements. The values you assign to each statement pair must increase relative to the previous pair's values. This is shown in the segmented graph in "RED Drop Profiles Overview" on page 3.

Range: 0 through 100 percent

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

Related Documentation

- RED Drop Profiles Overview on page 3
- Configuring RED Drop Profiles on page 9
- Understanding CoS Tail Drop Profiles
- Configuring CoS Tail Drop Profiles (CLI Procedure)

interpolate

Syntax

```
interpolate {
  drop-probability [values];
  fill-level [values];
}
```

Hierarchy Level [edit class-of-service drop-profiles *profile-name*]

Release Information Statement introduced before Junos OS Release 7.4.
Statement introduced before Junos OS 11.4 for EX Series switches.

Description Specify values for interpolating relationship between queue fill level and drop probability.

On EX Series switches, this statement is supported only on EX8200 standalone switches and EX8200 Virtual Chassis.

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

- See Configuring RED Drop Profiles on page 9.
- Understanding Junos OS CoS Components for EX Series Switches

red-buffer-occupancy

Syntax

```
red-buffer-occupancy {
  weighted-averaged [ instant-usage-weight-exponent exponent-value ];
}
```

Hierarchy Level

[edit chassis fpc *slot-number* pic *pic-number*],
[edit chassis lcc *number* fpc *slot-number* pic *pic-number*]

Release Information

Statement introduced in Junos OS Release 8.3.

Description

Configure the IQ PIC to base random early detection (RED) queue management on a *simple moving average* buffer occupancy calculation. If you do not include this statement, the IQ PIC bases RED on an *instantaneous* buffer occupancy value. As an option, you can specify that the IQ PIC bases RED on a *weighted moving average* of buffer occupancy values.

If you configure this feature on a channelized OC12 intelligent queuing (IQ) PIC, the PIC reboots.

Options

weighted-averaged—Configure the IQ PIC to base RED processing on a simple moving average of instantaneous buffer occupancy values instead of an instantaneous buffer occupancy.

instant-usage-weight-exponent *exponent-value*—(Optional) Specify the integer to be used as the negative exponent of 2 to express a weight value. The PIC performs weighted RED (WRED) by based on a calculation of average buffer occupancy that applies the specified weight value to the instantaneous buffer occupancy and then factors the weighted value into the calculation of average buffer occupancy. Valid exponent range is from 1 through 31 (weight values from 2^{-1} through 2^{-31}). If you do not specify this option, the default exponent value is 0, which results in a weight value of 2^{0} = 1. With a weight value of 1, the calculation of weighted average buffer occupancy yields the same value as the simple average buffer occupancy.

NOTE: You can specify an exponent value greater than 31, and the value displays in the output of show commands. However, the PIC replaces the out-of-range value with the *operational* value of 31, which results in a weight value of $2^{-31} = 1 / 2^{31}$ = 0.00000000046566128730773925781 25.

Required Privilege Level

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

Related Documentation

- Configuring Weighted RED Buffer Occupancy on page 9
- Example: Configuring Weighted RED Buffer Occupancy on page 14

PART 3

Index

Index

T

W