JUNIPER NETWORKS

Junos® OS

CoS Scheduling Feature Guide for Subscriber Services

Release
14.1

Published: 2014-04-25

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

Junos® OS CoS Scheduling Feature Guide for Subscriber Services
14.1

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:

- MX Series

Using the Examples in This Manual

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

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

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

Merging a Full Example

To merge a full example, follow these steps:

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

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

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

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

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

Merging a Snippet

To merge a snippet, follow these steps:

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

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

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

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

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

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

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

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

Documentation Conventions

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

Table 1: Notice Icons

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

Table 2 on page xv 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**

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

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

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

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

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

Convention	Description	Examples
> (bold right angle bracket)	Separates levels in a hierarchy of menu selections.	In the configuration editor hierarchy, select **Protocols>Ospf**.

Documentation Feedback

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

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

Requesting Technical Support

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

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

Self-Help Online Tools and Resources

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

- Find CSC offerings: http://www.juniper.net/customers/support/
- Search for known bugs: http://www2.juniper.net/kb/
- Find product documentation: http://www.juniper.net/techpubs/
- Find solutions and answer questions using our Knowledge Base: http://kb.juniper.net/
- Download the latest versions of software and review release notes: http://www.juniper.net/customers/csc/software/

- Search technical bulletins for relevant hardware and software notifications: http://kb.juniper.net/InfoCenter/
- Join and participate in the Juniper Networks Community Forum: http://www.juniper.net/company/communities/
- Open a case online in the CSC Case Management tool: http://www.juniper.net/cm/

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

Opening a Case with JTAC

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

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

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

PART 1

Overview

CHAPTER 1

CoS in Subscriber Access Networks

CoS for Subscriber Access Overview

This topic describes class-of-service (CoS) functionality for dynamic subscriber access.

Junos CoS enables you to divide traffic into classes and offer various levels of throughput and packet loss when congestion occurs. This functionality allows packet loss to happen according to rules that you configure. The Junos CoS features provide a set of mechanisms that you can use to provide differentiated services when best-effort traffic delivery is insufficient.

In a subscriber access environment, service providers want to provide video, voice, and data services over the same network for subscribers. Subscriber traffic is delivered from the access network, through a router, through a switched Ethernet network, to an Ethernet digital subscriber line access multiplexer (DSLAM). The DSLAM forwards the subscriber's traffic to the residential gateway over a digital subscriber line (DSL). An MX Series router that is installed in a subscriber access network as an edge router can perform subscriber management functions that include subscriber identification and per-subscriber CoS.

In a subscriber access network, a subscriber is an authenticated user—a user that has logged in to the access network at a subscriber interface and then been verified by the configured authentication server and subsequently granted initial CoS services. Subscribers can be identified statically or dynamically. In this network, subscribers are mapped to VLANs, demux, or PPPoE interfaces.

You can configure the router to provide *hierarchical scheduling* or *per-unit scheduling* for subscribers:

- Hierarchical CoS enables you to apply traffic scheduling and queuing parameters (which can include a delay-buffer bandwidth) and packet transmission scheduling parameters (which can include buffer management parameters) to an individual subscriber interface rather than to all interfaces configured on the port. Hierarchical CoS enables you to dynamically modify queues when subscribers require services.

- Per-unit scheduling enables one set of output queues for each logical interface configured under the physical interface. In per-unit scheduling configurations, each Layer 3 scheduler node is allocated a dedicated set of queues.

Because the interface sets corresponding to VLANs using agent-circuit-identifier information are created dynamically, you can apply CoS attributes, such as shaping, at the household level. You must set and define the CoS policy for the agent-circuit-identifier virtual VLAN interface set using the dynamic profile for the agent-circuit-identifier interface set (not the subscriber profile). CoS on dynamic VLANs includes support for level 3 or level 2 scheduler nodes for a dynamic interface set. You can also configure a traffic-control profile and a remaining traffic-control profile for a dynamic interface set. CoS on dynamic VLANs enables you to configure a dynamic scheduler map for a traffic-control profile that is used by a dynamic interface set. In this case, the dynamic scheduler map must use the unique ID format.

Related Documentation

- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- Configuring Per-Unit Scheduling in a Dynamic Profile for Subscriber Access on page 53

Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers

This topic describes hierarchical CoS on MX Series routers running Modular Port Concentrator/Modular Interface Card (MPC/MIC) or Enhanced Queuing Dense Port Concentrator (EQ DPC) interfaces.

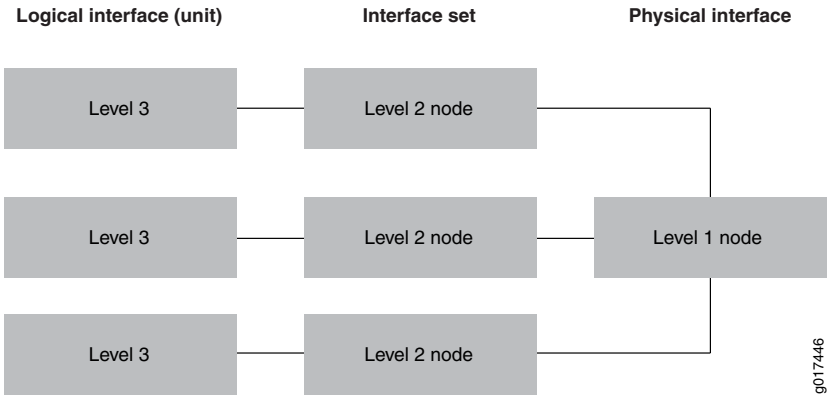
Hierarchical CoS enables you to apply traffic scheduling and queuing parameters and packet transmission scheduling parameters to an individual subscriber interface rather than to all interfaces configured on the port. Hierarchical CoS is supported on MX Series routers with either EQ DPCs or MPC/MICs installed.

On Juniper Networks MX Series routers, MPC/MIC and EQ DPC interfaces support a four-level CoS scheduling hierarchy that, when fully configured, consists of the physical interface (level 1), an interface set or underlying interface (level 2), one or more logical interfaces (level 3), and one or more queues (level 4). Although all CoS scheduling hierarchies are four-level, level 1 is always the physical interface and level 4 is always the queue. Hierarchical scheduling configurations consist of the type of interfaces you configure; for example, a logical interface or an interface set and where those interfaces reside in the scheduling hierarchy, either level 2 or level 3. Because many hierarchical scheduling configurations are possible, we use the terms *two-level hierarchical scheduling* and *three-level hierarchical scheduling* in this discussion.

Two-Level Hierarchical Scheduling

Two-level hierarchical scheduling limits the number of hierarchical levels in the scheduling hierarchy to two (level 2 and level 3) as shown in Figure 1 on page 5. In this configuration, interface sets are not configured and only the logical interfaces have traffic-control profiles. Two-level hierarchical scheduling is supported on MX Series routers running either MPC/MIC or EQ DPC interfaces.

Figure 1: Two-Level Hierarchical Scheduling



In a two-level scheduling hierarchy, all logical interfaces and interface sets share a single level 2 node; no hierarchical relationship is formed.

You control two-level hierarchical scheduling by including the **maximum-hierarchy-levels** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement:

- When the **maximum-hierarchy-levels** option is not set, interface sets can be at either level 2 or level 3, depending on whether the member logical interfaces within the interface set have a traffic-control profile.
- If any member logical interface has a traffic-control profile, then the interface set is always a level 2 CoS scheduler node.
- If no member logical interface has a traffic-control profile, the interface set is always a level 3 CoS scheduler node.
- If the **maximum-hierarchy-levels** option is set, then the interface set can only be at level 3; it cannot be at level 2. In this case, if you configure a level 2 interface set, you generate Packet Forwarding Engine errors.

Table 3 on page 5 summarizes the interface hierarchy and the CoS scheduler node levels for two-level hierarchical scheduling.

Table 3: Two-Level Hierarchical Scheduling-Interface Hierarchy Versus Scheduling Nodes

Level 1	Level 2	Level 3	Level 4
Physical interface	–	Logical interfaces	One or more queues

Table 3: Two-Level Hierarchical Scheduling-Interface Hierarchy Versus Scheduling Nodes *(continued)*

Level 1	Level 2	Level 3	Level 4
Physical interface	–	Interface set	One or more queues
Physical interface	–	Logical interfaces	One or more queues

To configure two-level hierarchical scheduling, include the **hierarchical-scheduler** statement at the **[edit interfaces *interface-name*]** hierarchy level. You can optionally include the **maximum-hierarchy-levels** option. If you choose to set this option, the only supported value is 2.

```
[edit interfaces]
xe-2/0/0 {
  hierarchical-scheduler {
    maximum-hierarchy-levels 2;
  }
}
```

Three-Level Hierarchical Scheduling

Three-level hierarchical scheduling is supported only on MX Series routers running MPC/MIC interfaces. Three-level hierarchical scheduling has up to eight class of service queues. You can configure many different three-level scheduling hierarchies, depending on the location of the interface set and the use of underlying interfaces. In all variations, the physical interface is a level 1 CoS scheduler node and the queues reside at level 4.

NOTE: **Three-level hierarchical scheduling is supported only on subscriber interfaces and interface sets running over aggregated Ethernet interfaces on MPC/MIC interfaces in MX Series routers.**

When you use three-level hierarchical scheduling, interface sets can reside at either level 2 or level 3. You can also configure an underlying logic interface at level 2 and a logical interface at level 3. Table 4 on page 6 summarizes the most common cases of the interface hierarchy and the CoS scheduler node levels for three-level hierarchical scheduling.

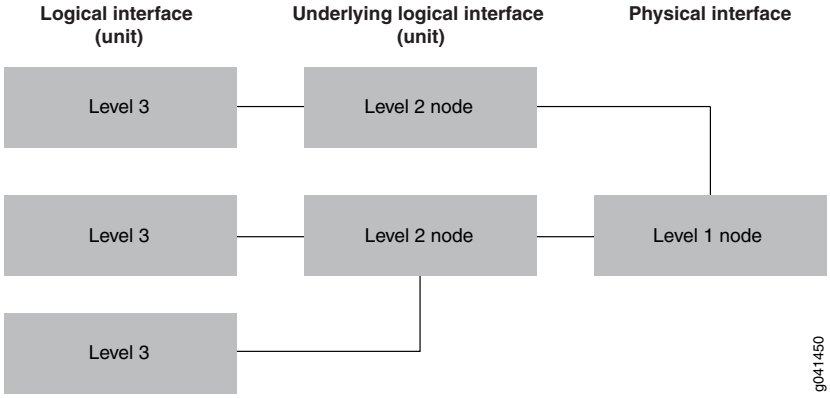
Table 4: Three-Level Hierarchical Scheduling-Interface Hierarchy Versus CoS Scheduling Node Levels

Level 1	Level 2	Level 3	Level 4
Physical interface	Interface set	Logical interfaces	One or more queues
Physical interface	Logical interface	Interface set	One or more queues
Physical interface	Underlying logical interface	Logical interfaces	One or more queues

In three-level hierarchical scheduling, the CoS scheduler nodes at level 1, level 2, and level 3 form a hierarchical relationship; this differs from two-level hierarchical scheduling where no hierarchical relationship is formed.

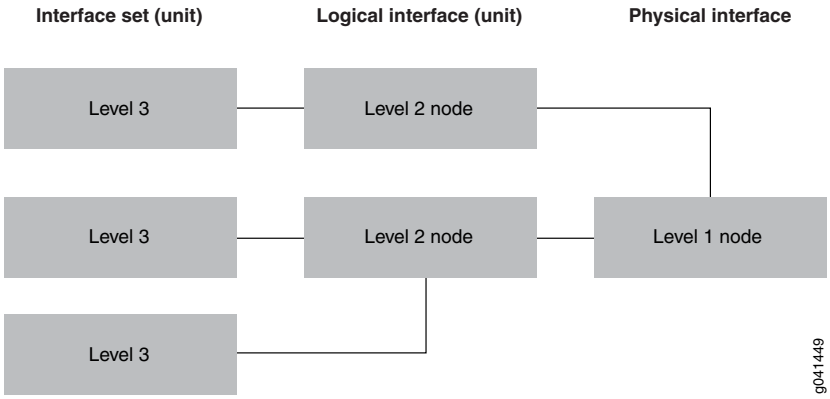
With a three-level hierarchical scheduling, logical interfaces can reside at level 2, or they can reside at level 3, if the logical interface at level 2 is an underlying logical interface. This is shown in Figure 2 on page 7.

Figure 2: Three-Level Hierarchical Scheduling: Logical Interfaces at Level 3 with Underlying Logical Interfaces at Level 2



Another possible configuration for three-level hierarchical scheduling is shown in Figure 3 on page 7. In this configuration, the logical interfaces are located at level 2 and the interface sets are located at level 3.

Figure 3: Three-Level Hierarchical Scheduling: Logical Interfaces at Level 2 with Interface Sets at Level 3



To configure three-level hierarchical scheduling, include the **implicit-hierarchy** option at the **[edit interfaces *interface-name* hierarchical-scheduler]** hierarchy level.

```
[edit interfaces]
xe-2/0/0 {
  hierarchical-scheduler {
    implicit-hierarchy;
  }
}
```

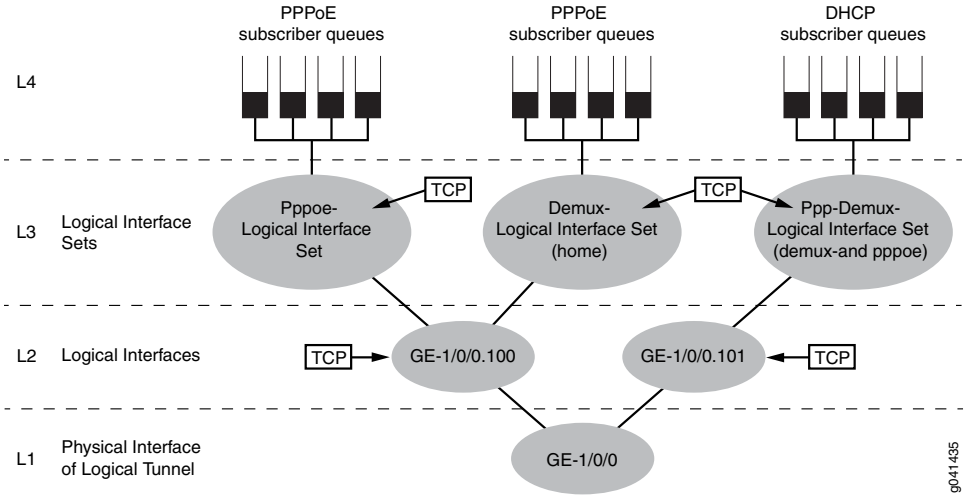
Interface Hierarchy Versus CoS Hierarchy

An interface hierarchy and a CoS scheduling hierarchy are distinctly different. Interface hierarchy refers to the relationship between the various interfaces; for example, the relationship between logical interfaces and an interface set, the relationship between a logical interface and an underlying logical interface, or the relationship between the physical interface and logical interface. CoS scheduling hierarchy refers to the hierarchical relationship between the CoS scheduler nodes. In two-level hierarchical scheduling, no hierarchy is formed between the CoS scheduler nodes; all logical interfaces and interface sets share a single level 2 scheduler node. However, when you use the **implicit-hierarchy** option for three-level hierarchical scheduling, the CoS scheduler nodes form a scheduling hierarchy.

Figure 4 on page 8 and Figure 5 on page 9 provide two scenarios for this discussion. Figure 4 on page 8 shows an interface hierarchy where a Gigabit Ethernet interface (GE-1/0/0) is the physical interface. Two logical interfaces (GE-1/0/0.100 and GE-1/0/0.101) are configured on the physical interface:

- Logical interface GE-1/0/0.100 is a member of a PPPoE interface set and a Demux interface set.
- Logical interface GE-1/0/0.101 is a member of a demux interface set.

Figure 4: Logical Interfaces at Level 2 and Interface Sets at Level 3



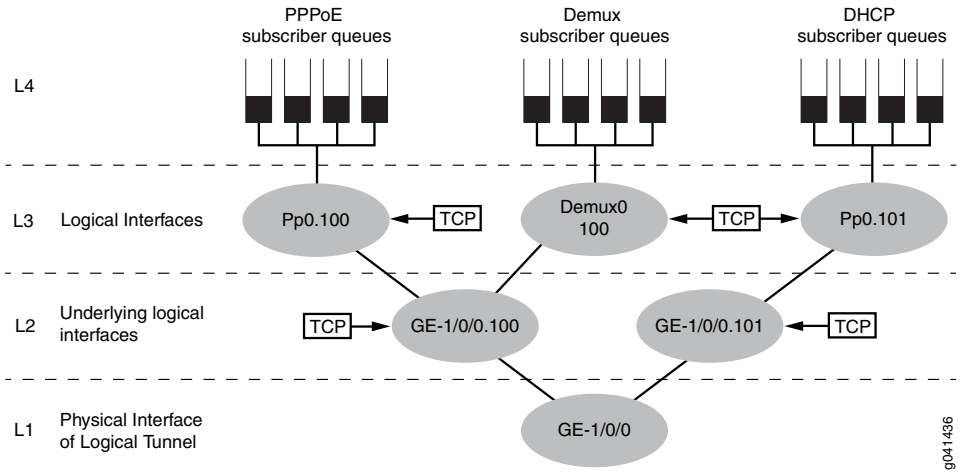
Each interface set has a dedicated queue. The CoS scheduler nodes at level 1 (physical interface), level 2 (underlying logical interfaces), and level 3 (interface sets) form a scheduling hierarchy.

To configure this scenario, you must include the **implicit-hierarchy** option under the **hierarchical-scheduler** statement on physical interface GE-1/0/0 and configure and apply traffic-control profiles on each interface set and underlying logical interface.

Figure 5 on page 9 shows an interface hierarchy where Gigabit Ethernet interface GE-1/0/0 is the physical interface. Three logical interfaces are configured:

- Two logical interfaces (Pp0.100 and Demux0.100) reside on the underlying logical interface GE-1/0/0.100.
- A third logical interface (Pp0.101) resides on the underlying logical interface GE-1/0/0.101.

Figure 5: Logical Interfaces at Level 3 and Underlying Logical Interfaces at Level 2



Each logical interface has a dedicated queue. The CoS scheduler nodes at level 1 (physical interface), level 2 (underlying logical interfaces), and level 3 (logical interfaces) form a scheduling hierarchy.

To configure this scenario, you must include the **implicit-hierarchy** option under the **hierarchical-scheduler** statement on physical interface GE-1/0/0 and configure and apply traffic-control profiles on each logical interface and underlying logical interface.

You can configure many different three-level scheduling hierarchies; Figure 4 on page 8 and Figure 5 on page 9 present just two possible scenarios. Table 4 on page 6 summarizes the possible interface locations and CoS scheduler nodes.

Related Documentation

- *Configuring Hierarchical Schedulers for CoS*
- Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links on page 86
- Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface on page 87
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 34
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

CHAPTER 2

CoS and Specific Interface Types

CoS for Aggregated Ethernet Subscriber Interfaces Overview

You can apply static or dynamic hierarchical CoS to a scheduler node at the aggregated Ethernet logical interface, its underlying physical interface, or an interface set.

Guidelines for Configuring CoS for Aggregated Ethernet Subscribers

When you configure CoS for aggregated Ethernet interfaces, consider the following guidelines:

- Configure the aggregated Ethernet logical interface over two physical interfaces capable of performing hierarchical scheduling.
- For VLAN subscriber interfaces over aggregated Ethernet, you must enable link protection on the aggregated Ethernet interface for hierarchical CoS to operate.
- Link protection is not required for IP or demux subscriber interfaces over aggregated Ethernet. We recommend that you enable targeted distribution on the demux interface to provide accurate hierarchical scheduling for these links.
- Keep the following guidelines in mind when configuring interface sets of aggregated Ethernet interfaces:
  - Sets of aggregated Ethernet interfaces are supported on MPC/MIC interfaces on MX Series routers only.
  - The supported logical interfaces for aggregated Ethernet in an interface set include VLAN demux interfaces, IP demux interfaces, and PPPoE logical interfaces over VLAN demux interfaces.

- The link membership list and scheduler mode of the interface set are inherited from the underlying aggregated Ethernet interface over which the interface set is configured.
- When an aggregated Ethernet interface operates in link protection mode, or if the scheduler mode is configured to replicate member links, the scheduling parameters of the interface set are copied to each of the member links.
- If the scheduler mode of the aggregated Ethernet interface is set to scale member links, the scheduling parameters are scaled based on the number of active member links and applied to each of the aggregated interface member links.

Related Documentation

- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- For hardware requirements, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- For configuration instructions, see Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links on page 86 and Configuring an Interface Set of Subscribers in a Dynamic Profile on page 85
- For additional information about subscribers over aggregated Ethernet, see *Static or Dynamic Demux Subscriber Interfaces over Aggregated Ethernet Overview*, *Distribution of Demux Subscribers in an Aggregated Ethernet Interface*, and *Static and Dynamic VLAN Subscriber Interfaces over Aggregated Ethernet Overview*.

CoS for PPPoE Subscriber Interfaces Overview

You can configure CoS functionality for static and dynamic PPPoE subscriber interfaces configured on Gigabit Ethernet Intelligent Queuing 2 (IQ2) and Ethernet Enhanced IQ2 (IQ2E) PICs on the M120 and M320 routers, and on MPCs on the MX Series 3D Universal Edge Router.

For all supported hardware platforms, you can attach an output traffic-control profile that contains basic shaping and scheduling properties directly to a PPPoE interface. In this type of scenario, you can use each PPPoE interface to represent a household and shape all of the household traffic to an aggregate rate. Each forwarding class is mapped to a queue, and represents one type of services provided to a household customer.

Both the IQ2E PIC and MPC support hierarchical scheduling functionality that is not available on the IQ2 PIC. To shape customer or DSLAM traffic at different levels of the PPPoE interface hierarchy, you can attach traffic-control profiles to interface sets that contain PPPoE members.

MPCs support subscriber interfaces with PPPoE encapsulation over aggregated Ethernet interfaces. These PPPoE subscriber interfaces are configured over VLAN demux interfaces, which are also configured over Aggregated Ethernet interfaces.

You can configure 802.3ad link aggregation group (LAG) stateful port and dense port concentrator (DPC) redundancy. This provides targeted distribution of non-replicated

(stacked) PPPoE or IP dlemux links over VLAN demux links, which in turn are over an aggregated Ethernet (AE) logical interface. Service providers with PPPoE or IP demux interfaces for CoS configurations can provide DPC and port redundancy to subscribers.

NOTE: **For static PPPoE underlying logical interfaces, use PPPoE interface sets.**

Related Documentation

- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface on page 87
- *CoS on Enhanced IQ2 PICs Overview*

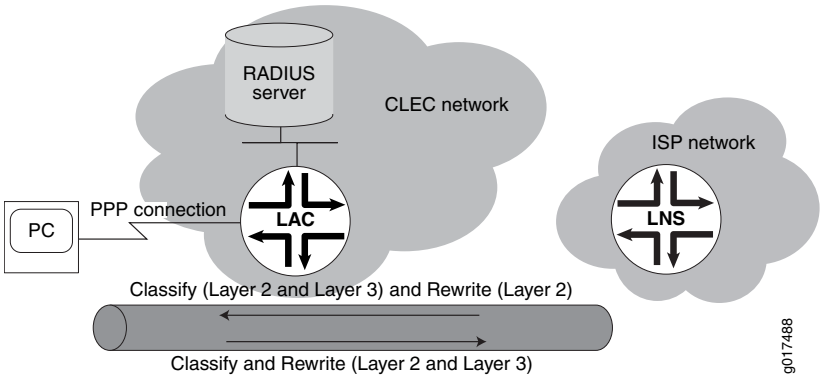
CoS for L2TP LAC Subscriber Interfaces Overview

You can apply CoS to the Layer 2 Tunnel Protocol (L2TP) access concentrator (LAC) component.

In Layer 2 Tunnel Protocol (L2TP) configurations, IP and L2TP headers are added to packets arriving at a PPP subscriber interface on the L2TP access concentrator (LAC) before being tunneled to the L2TP network server (LNS). You can manage the IP header by configuring classifiers and rewrite-rules that transfer the ToS (Type of Service) value or the 802.1p value from the *inner* IP header to the *outer* IP header of the L2TP packet.

Figure 6 on page 13 shows the classifier and rewrite rules that you can configure from the LAC to the LNS, and from the LNS to the LAC.

Figure 6: CoS Configuration for L2TP LAC Topology



- Traffic from LAC to LNS on page 14
- LAC Tunnels: Traffic from LNS to LAC on page 14

Traffic from LAC to LNS

To set the ToS value or the 802.1p value on the inner IP header, you can configure both fixed and behavior aggregate (BA) classifiers for subscribers at Layer 2 or Layer 3 of the network.

Table 5 on page 14 lists the configuration options for applying classifiers to a subscriber interface on an ingress LAC tunnel.

Table 5: Ingress LAC Tunnel Classifier Options

Classifier	Subscriber Interface
Fixed	Either of the following: • PPP interface • Underlying VLAN interface
Layer 2	Either of the following: • PPP interface • Underlying VLAN interface
Layer 3	Family of PPP interfaces

You cannot configure a Layer 2 and fixed classifier together.

The behavior of the Layer 2 and Layer 3 classifiers depends on the configuration. For example, a Layer 3 classifier for a family of PPP interfaces overrides a Layer 2 classifier configured at the PPP interface, except for the unknown packets and control packets.

If you do not configure a classifier for Layer 2, the system applies the default Layer 3 classifier so that tunneled and terminated subscribers have the same behavior. To prevent unknown packets and control packets from being discarded, the system assigns them to the best-effort forwarding class.

For egress tunnels, you configure rewrite rules at the PPP interface to set the ToS or 802.1p value of the outer IP header. Rewrite rules are applied accordingly to the forwarding class, packet loss priority (PLP), and code point.

LAC Tunnels: Traffic from LNS to LAC

On a LAC, mapping the inner IP header to the outer IP header of the L2TP packet depends on the classifier and rewrite-rule configurations. For example, Table 6 on page 15 lists the values for the classifier and rewrite rules for a VLAN interface. For assured forwarding, the inner 802.1p value (**ob001**) is classified with the assured-forwarding class and low loss priority at the ingress interface. Based on the assured-forwarding class and low loss priority in the rewrite rule, the ToS value in the outer IP header is set to **ob001**.

Table 6: Sample Result

Inner .1p Value	Forwarding Class	Loss Priority	Code Point	Outer ToS Value
ob000	best-effort	low	000	ob000
ob001	assured-forwarding	low	001	ob001
ob101	expedited-forwarding	low	101	ob101
ob111	network-control	low	11	ob111

Related Documentation

- Configuring Dynamic CoS for an L2TP LAC Tunnel on page 88

CoS for L2TP LNS Inline Services Overview

You can apply hierarchical scheduling and per-session shaping to Layer 2 Tunnel Protocol (L2TP) network server (LNS) inline services using a static or dynamic CoS configuration.

This feature is supported on MIC and MPC interfaces on MX240, MX480, and MX960 routers.

- Guidelines for Applying CoS to the LNS on page 15
- Hardware Requirements for Inline Services on the LNS on page 16

Guidelines for Applying CoS to the LNS

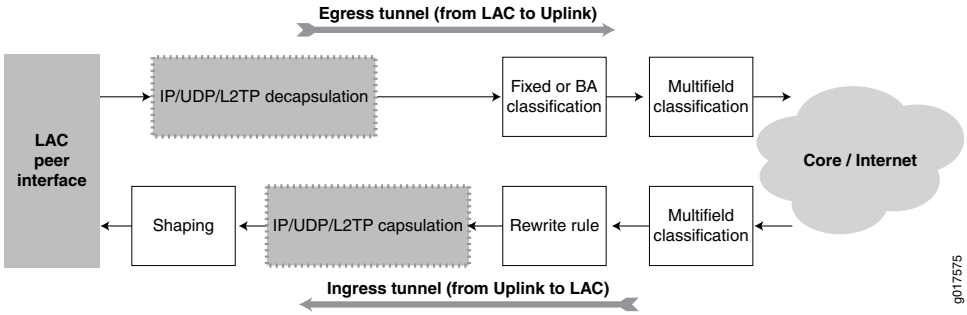
In L2TP configurations, IP, UDP, and L2TP headers are added to packets arriving at a PPP subscriber interface on the L2TP access concentrator (LAC) before being tunneled to the LNS.

When a service interface is configured for an L2TP LNS session, it has an *inner* IP header and an outer IP header. You can configure CoS for an LNS session that corresponds to the inner IP header only. The *outer* IP header is used for L2TP tunnel processing only.

However, we recommend that you configure classifiers and rewrite-rules to transfer the ToS (type of service) value from the inner IP header to the outer IP header of the L2TP packet.

Figure 7 on page 16 shows the classifier and rewrite rules that you can configure on an LNS inline service.

Figure 7: Processing of CoS Parameters in an L2TP LNS Inline Service



By default, the shaping calculation on the service interface includes the L2TP encapsulation. If necessary, you can configure additional adjustments for downstream ATM traffic from the LAC or differences in Layer 2 protocols.

Hardware Requirements for Inline Services on the LNS

Hierarchical scheduling for L2TP LNS inline services is supported on MIC and MPC interfaces only. The services that you can configure depend on the hardware combination. Table 7 on page 16 lists the supported inline services and peer interfaces for each MIC and MPC combination.

Table 7: Hardware Requirements for L2TP LNS Inline Services

MPC Module	Inline Service Support–With Per-Session Shaping	Inline Service Support–Without Per-Session Shaping
MX-MPC1-3D MX-MPC2-3D	No	Yes
MX-MPC1-3D-Q MX-MPC2-3D-Q MX-MPC2-3D-EQ MX80	Yes	Yes
MPC-3D-16XGE-SFPP	No	No

Related Documentation

- *Configuring Static CoS for an L2TP LNS Inline Service*
- Configuring Dynamic CoS for an L2TP LNS Inline Service on page 89

CoS for Interface Sets of Subscribers Overview

Interface sets enable service providers to group logical interfaces so they can apply CoS parameters to all of the traffic in the group.

Interface sets are beneficial for various scenarios in a subscriber access network. For example, you can use an interface set to configure a local loop with a small number of subscribers. Interface sets are also useful for grouping a large number of subscribers into a particular service class or for defining traffic engineering aggregates for DSLAMs.

- Guidelines for Configuring Dynamic Interface Sets in a Subscriber Access Network on page 17

Guidelines for Configuring Dynamic Interface Sets in a Subscriber Access Network

Interface sets enable service providers to group logical interfaces so they can apply CoS parameters to all of the traffic in the group.

Interface sets are beneficial for various scenarios in a subscriber access network. For example, you can use an interface set to configure a local loop with a small number of subscribers. Interface sets are also useful for grouping a large number of subscribers into a particular service class or for defining traffic engineering aggregates for DSLAMs.

When configuring interface sets for subscriber access, keep the following guidelines in mind:

- You can configure interface sets of VLAN demux, PPPoE, or demux interfaces over aggregated Ethernet interfaces.
- An interface can only belong to one interface set. If you try to add the same interface to different interface sets, the commit operation fails.
- You configure the interface set and the traffic scheduling and shaping parameters in a dynamic profile. However, you must apply the traffic-control profile to the interface set in the static **[edit class-of-service]** hierarchy.

  NOTE: **This rule applies to all interface sets except ACI sets.**

- The **$junos-interface-set-name** predefined variable is available only for RADIUS Accept messages; change of authorization (CoA) requests are not supported.
- The **$junos-svlan-interface-set-name** predefined variable locally generates an interface set name for use by dual-tagged VLAN interfaces based on the outer tag of the dual-tagged VLAN. The format of the generated variable is ***physical_interface_name - outer_VLAN_tag***. For example, an aggregated Ethernet interface “ae0,” with a dual-tagged VLAN interface that has an outer tag of “111,” results in a **$junos-svlan-interface-set-name** dynamic variable of “ae0-111”. Similarly, a non-aggregated Ethernet interface of ge-1/1/0, with the same dual-tagged VLAN interface that has an outer tag of “111,” results in a **$junos-svlan-interface-set-name** dynamic variable of “ge-1/1/0-111”.

- The **$junos-tagged-vlan-interface-set-name** predefined variable locally generates an interface set name used for grouping logical interfaces stacked over logical stacked VLAN demux interfaces for either a 1:1 (dual-tagged; individual client) VLAN or N:1 (single tagged; service) VLAN. The format of the generated variable differs with VLAN type as follows:
  - Dual-tagged (client) VLAN—***physical_interface_name - outer_VLAN_tag - inner_VLAN_tag***. For example, an aggregated Ethernet interface "ae0," with a dual-tagged VLAN interface that has an outer tag of "111" and an inner tag of "200," results in a **$junos-tagged-vlan-interface-set-name** dynamic variable of "ae0-200-111". Similarly, a non-aggregated Ethernet interface of ge-1/1/0, with the same dual-tagged VLAN interface that has an outer tag of "111" and an inner tag of "200," results in a **$junos-tagged-vlan-interface-set-name** dynamic variable of "ge-1/1/0-200-111".
  - Single tagged (service) VLAN—***physical_interface_name - VLAN_tag***. For example, an aggregated Ethernet interface "ae0," with an N:1 VLAN using the single tag of "200," results in a **$junos-tagged-vlan-interface-set-name** dynamic variable of "ae0-200". Similarly, a non-aggregated Ethernet interface of ge-1/1/0, with the same N:1 VLAN using the single tag of "200," results in a **$junos-tagged-vlan-interface-set-name** dynamic variable of "ge-1/1/0-200".
- All dynamic demux, dual-tagged VLAN logical interfaces with the same outer VLAN tag and physical interface are assigned to the same interface set and all CoS values provisioned with the dynamic profile are applied to the interfaces that are part of the set.
- The interface set name must be explicitly referenced in the CoS configuration as part of the static configuration outside of the dynamic profile. The CoS configuration is static and the interface set name must be statically referenced.

NOTE: This rule applies to all interface sets except ACI sets.

- RADIUS can return an *access-accept* message under certain conditions. A configured RADIUS VSA for the interface set name takes precedence over the locally generated variable on the router. This means that if the interface-set-name VSA is configured on RADIUS, the router continues to use this variable instead of the locally generated value from the dynamic variable.
- Sets of aggregated Ethernet interfaces are supported on MPC/MIC interfaces on MX Series routers only.
- The supported interface stacks for aggregated Ethernet in an interface set include VLAN demux interfaces, IP demux interfaces, and PPPoE logical interfaces over VLAN demux interfaces.
- The link membership list and scheduler mode of the interface set are inherited from the underlying aggregated Ethernet interface over which the interface set is configured.

- When an aggregated Ethernet interface operates in link protection mode, or if the scheduler mode is configured to replicate member links, the scheduling parameters of the interface set are copied to each of the member links.
- If the scheduler mode of the aggregated Ethernet interface is set to scale member links, the scheduling parameters are scaled based on the number of active member links and applied to each of the aggregated interface member links.

Related Documentation

- Configuring an Interface Set of Subscribers in a Dynamic Profile on page 85
- Example: Configuring a Dynamic Service VLAN Interface Set of Subscribers in a Dynamic Profile on page 143

CHAPTER 3

CoS and RADIUS

Subscriber Interfaces That Provide Initial CoS Parameters Dynamically Obtained from RADIUS

You can configure interface-specific CoS parameters that the router obtains when subscribers log in at appropriately configured static or dynamic subscriber interfaces. This feature is supported only for interfaces on Enhanced Queuing Dense Port Concentrators (EQ DPCs) in MX Series 3D Universal Edge Routers.

To configure a dynamic profile to provide initial CoS Services, make sure you understand the following concepts:

Dynamic Configuration of Initial CoS in Access Profiles

When a router interface receives a join message from a DHCP subscriber, the Junos OS applies the values configured in the dynamic profile associated with that router interface. A dynamic profile that is activated through its association with a subscriber interface is known as an *access dynamic profile*. You can associate a dynamic profile with a subscriber interface on the router by including statements at the **[edit dynamic-profiles *profile-name* class-of-service interfaces]** hierarchy level.

The Junos OS supports predefined variables for obtaining a scheduler-map name and traffic-shaping parameters from the RADIUS authentication server and predefined variables for obtaining a scheduler name and scheduler parameters from the RADIUS authentication server. When a client authenticates over a router interface associated

with the access dynamic profile, the router replaces the predefined variables with interface-specific values obtained from the RADIUS server.

NOTE: To associate dynamically configured initial CoS features with a subscriber interface, reference *Junos OS predefined variables*—and not *user-defined variables*—in an *access* dynamic profile for that interface.

Predefined Variables for Dynamic Configuration of Initial Traffic Shaping

You can configure an access dynamic profile that provides initial traffic-shaping parameters when a subscriber logs in. The Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.

If you define the Juniper Networks authentication and authorization VSA for CoS traffic-shaping parameter values (attribute number 26–108) on the RADIUS authentication server, the RADIUS server includes the values in RADIUS Access-Accept messages it sends to the router when a subscriber successfully authenticates over the interface.

To provide an initial scheduler map name and traffic shaping parameters obtained from the RADIUS authentication server when a subscriber logs in, reference the Junos OS predefined variables for CoS listed in Table 8 on page 22 in an access dynamic profile associated with the subscriber interface.

Table 8: CoS Predefined Variables for Scheduler Map and Traffic Shaping

Variable	Description
$junos-cos-scheduler-map	Scheduler-map name to be dynamically configured in a traffic-control profile in the access dynamic profile when a subscriber logs in. **NOTE:** The scheduler map referenced by the scheduler-map statement can be defined dynamically (at the **[edit dynamic-profiles *profile-name* class-of-service scheduler-maps]** hierarchy level) or statically (at the **[edit class-of-service scheduler-maps]** hierarchy level).
$junos-cos-shaping-rate	Shaping rate to be dynamically configured in a traffic-control profile in the access dynamic profile when a subscriber logs in. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.
$junos-cos-guaranteed-rate	Guaranteed rate to be dynamically configured in a traffic-control profile in the access dynamic profile when a subscriber logs in. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.
$junos-cos-delay-buffer-rate	Delay-buffer rate to be dynamically configured in a traffic-control profile in the access dynamic profile when a subscriber logs in. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.

Predefined Variables for Dynamic Configuration of Initial Scheduling and Queuing

You can configure an access dynamic profile that provides initial traffic-shaping parameters when a subscriber logs in. The Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.

If you define the Juniper Networks authentication and authorization VSA for CoS scheduling and queuing parameter values (attribute number 26–146) on the RADIUS authentication server, the RADIUS server includes the values in RADIUS Access-Accept messages it sends to the router when a subscriber successfully authenticates over the interface.

To provide an initial scheduler name and scheduler and queuing parameters obtained from the RADIUS authentication server when a subscriber logs in, reference the Junos OS predefined variables listed in Table 9 on page 23 in an access dynamic profile associated with the subscriber interface.

Table 9: CoS Predefined Variables for Scheduling and Queuing

Variable	Description
$junos-cos-scheduler	Name of a scheduler to be dynamically configured in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.
$junos-cos-scheduler-transmit-rate	Transmit rate to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.
$junos-cos-scheduler-bs	Buffer size, as a percentage of total buffer, to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.
$junos-cos-scheduler-pri	Packet-scheduling priority value to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached.
$junos-cos-scheduler-dropfile-low	Name of the drop profile for RED for loss-priority level **low** to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached. **NOTE:** The drop profile must be configured statically (at the **[edit** class-of-service schedulers ***scheduler-name*** **drop-profiles]** hierarchy level) for loss-priority **low**.

Table 9: CoS Predefined Variables for Scheduling and Queuing *(continued)*

Variable	Description
$junos-cos-scheduler-dropfile-medium-low	Name of the drop profile for RED for loss-priority level **medium-low** to be dynamically configured for the scheduler in the access dynamic profile. The Junos OS obtains this information from the RADIUS server when a subscriber authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached. **NOTE:** The drop profile must be configured statically (at the **[edit** class-of-service schedulers ***scheduler-name*** **drop-profiles]** hierarchy level).
$junos-cos-scheduler-dropfile-medium-high	Name of the drop profile for RED for loss-priority level **medium-high** to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached. **NOTE:** The drop profile must be configured statically (at the **[edit** class-of-service schedulers ***scheduler-name*** **drop-profiles]** hierarchy level).
$junos-cos-scheduler-dropfile-high	Name of the drop profile for RED for loss-priority level **high** to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached. **NOTE:** The drop profile must be configured statically (at the **[edit** class-of-service schedulers ***scheduler-name*** **drop-profiles]** hierarchy level).
$junos-cos-scheduler-dropfile-any	Name of the drop profile for RED for loss-priority level **any** to be dynamically configured for the scheduler in the access dynamic profile. You can configure a RADIUS authentication server to include this information in the Accept-Accept message when a subscriber successfully authenticates over the static or dynamic subscriber interface to which the access dynamic profile is attached. **NOTE:** The drop profile must be configured statically (at the **[edit** class-of-service schedulers ***scheduler-name*** **drop-profiles]** hierarchy level).

Related Documentation

- *Subscriber Activation and Service Management in an Access Network*
- *Dynamic Profiles Overview*
- *Dynamic Variables Overview*
- *Junos OS Predefined Variables*
- Configuring Initial CoS Parameters Dynamically Obtained from RADIUS on page 67
- Example: Configuring Initial CoS Parameters Dynamically Obtained from RADIUS on page 119

Changing CoS Services Overview

This topic describes how to provide CoS when subscribers dynamically upgrade or downgrade services in an access environment.

You can configure your network with an *access profile* that provides all subscribers with default CoS parameters when they log in. For example, all subscribers can receive a basic data service. By configuring the access profile with Junos OS predefined variables for RADIUS-provided CoS parameters, you also enable the service to be activated for those subscribers at login.

To enable subscribers to activate a service or upgrade to different services through RADIUS change-of-authorization (CoA) messages after login, configure a *service profile* that includes user-defined variables.

Types of CoS Variables Used in a Service Profile

You can configure variables for the following CoS parameters in a service profile:

- Shaping rate
- Delay buffer rate
- Guaranteed rate
- Scheduler map

For each CoS parameter, you must associate a RADIUS vendor ID. For each vendor ID, you must assign an attribute number and a tag. The tag is used to differentiate between values for different CoS variables when you specify the same attribute number for those variables. These values are matched with the values supplied by RADIUS during subscriber authentication. All of the values in the dynamic profile must be defined in RADIUS or none of the values are passed.

Optionally, you can configure default values for each parameter. Configuring default values is beneficial if you do not configure RADIUS to enable service changes. During service changes, RADIUS takes precedence over the default value that is configured.

Static and Dynamic CoS Configurations

Depending on how you configure CoS parameters in the access and service profiles, certain CoS parameters are replaced or merged when subscribers change or activate new services.

Static configuration is when you configure the scheduler map and schedulers in the static **[edit class-of-service]** hierarchy and reference the scheduler map in the dynamic profile. Dynamic configuration is when you configure the scheduler map and schedulers within the dynamic profile.

The CoS configuration also depends on whether you have enabled multiple subscribers on the same logical interface using the **aggregate-clients** statements in the dynamic profile referenced by DHCP. When you specify the **aggregate-clients replace** statement, the scheduler map names are replaced. In both cases, if the length of the scheduler map

name exceeds 128 characters, subscribers cannot log in. When you specify the **aggregate-clients merge** statement, the scheduler map names specified in the dynamic profile are appended.

BEST PRACTICE: To improve CoS performance in IPv4, IPv6, and dual-stack networks, we recommend that you use the **aggregate-clients replace** statement rather than the **aggregate-clients merge** statement.

Scenarios for Static and Dynamic Configuration of CoS Parameters

Table 10 on page 26 lists the scenarios for static and dynamic configuration of CoS parameters in access profiles and service profiles at subscriber login. The table also lists the behavior for each configuration for service activation and service modification using RADIUS CoA messages.

Table 10: CoS Services and Variables

Scenario	Static CoS Configuration (Single Subscriber)	Dynamic CoS Configuration (Single Subscriber)	Dynamic CoS Configuration (Multiple Subscribers Enabled on a Logical Interface with the aggregate-clients merge Statement)	Dynamic CoS Configuration (Multiple Subscribers Enabled on a Logical Interface with the aggregate-clients replace Statement)
Subscriber login	• Configure RADIUS values or default values for all parameters in access profile • Configure scheduler map in **edit class-of-service** hierarchy and reference in access profile	• Configure RADIUS values or default values for all parameters in access profile • Configure scheduler map and schedulers in access profile	• Configure RADIUS values or default values for all parameters in access profile • Configure scheduler map and schedulers in access profile	• Configure RADIUS values or default values for all parameters in access profile • Configure scheduler map and schedulers in access profile
RADIUS CoA for service or variable change	Replaces the following parameters: • Delay buffer rate • Guaranteed rate • Scheduler map • Shaping rate	Replaces the following parameters: • Delay buffer rate • Guaranteed rate • Shaping rate • Scheduler map	Combines the values of the following parameters to their maximum scalar value: • Delay buffer rate • Guaranteed rate • Shaping rate Appends the scheduler map parameter	Replaces the following parameters: • Delay buffer rate • Guaranteed rate • Shaping rate • Scheduler map

Table 10: CoS Services and Variables *(continued)*

Scenario	Static CoS Configuration (Single Subscriber)	Dynamic CoS Configuration (Single Subscriber)	Dynamic CoS Configuration (Multiple Subscribers Enabled on a Logical Interface with the aggregate-clients merge Statement)	Dynamic CoS Configuration (Multiple Subscribers Enabled on a Logical Interface with the aggregate-clients replace Statement)
RADIUS CoA for service activation	Does not merge queues **NOTE:**In this case, use a similar configuration to the access profile, including the same name for the traffic-control-profile. During service activation, this configuration replaces the original configuration in the access profile.	Merge queues if the queue specified in the service profile is not already in use for the subscriber **NOTE:** Do not instantiate a CoA request using a service dynamic profile that is already in use on the same logical interface.	Merge queues if the queue specified in the service profile is not already in use for the subscriber **NOTE:** Do not instantiate a CoA request using a service dynamic profile that is already in use on the same logical interface.	Merge queues if the queue specified in the service profile is not already in use for the subscriber **NOTE:** Do not instantiate a CoA request using a service dynamic profile that is already in use on the same logical interface.

Related Documentation

- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- *Dynamic Profile Attachment to DHCP Subscriber Interfaces Overview*
- *RADIUS Attributes and Juniper Networks VSAs Supported by the AAA Service Framework*
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41

CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions Overview

To control bandwidth at a household level in a subscriber access network, you can apply RADIUS dynamic class of service (CoS) traffic-shaping attributes to a dynamic interface set and its member subscriber sessions when the subscriber sessions are authenticated. (The dynamic interface set itself does not go through the authentication process.)

A *household* is represented by either a dynamic interface set or a dynamic agent-circuit-identifier (ACI) interface set from which the subscriber sessions originate. For this feature, dynamic interface sets and dynamic ACI interface sets are mapped to Level 2 of the Junos OS CoS scheduler hierarchy, which enables you to use CoS traffic-shaping to shape the bandwidth at the household (interface set) level.

The *subscriber sessions*, also referred to as *subscriber interfaces* or *client sessions*, can be dynamic VLAN, PPPoE, or IP demultiplexing (IP demux) subscriber interfaces. The subscriber interfaces are mapped to Level 3 of the Junos OS CoS scheduler hierarchy.

Supported Network Configurations

Applying RADIUS dynamic CoS traffic-shaping attributes to a dynamic interface set and its member subscriber sessions is supported for the following network configurations:

- Dynamic IP demux subscriber interfaces (for DHCP subscribers) over either a dynamic interface set or a dynamic ACI interface set
- Dynamic PPPoE subscriber interfaces over either a dynamic interface set or a dynamic ACI interface set

Traffic-Control Profiles in Subscriber Interface Dynamic Profiles

To apply dynamic CoS traffic-shaping attributes to a dynamic interface set and its member subscriber sessions, you must define and attach the traffic-control profiles for *both* the dynamic interface set and the dynamic subscriber sessions within the dynamic profile for the subscriber interface.

At the **[edit dynamic-profiles *profile-name* class-of-service traffic-control-profiles]** hierarchy level in the dynamic profile, configure both of the following:

- Traffic-control profile for the dynamic VLAN, PPPoE, or IP demux subscriber interfaces
- Traffic-control profile for the dynamic interface set or dynamic ACI interface set to which the subscriber interfaces belong

RADIUS tag values for the Junos OS CoS traffic shaping predefined variables used in both traffic-control profiles must be in the 100s range, as described in "CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets" on page 73.

At the **[edit dynamic-profiles *profile-name* interfaces]** hierarchy level in the dynamic profile, use the **output-traffic-control-profile** statement to apply the traffic-control profiles to the dynamic subscriber interface and the dynamic interface set or dynamic ACI interface set.

CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets and Member Subscriber Sessions

The set of **$junos-cos-*parameter*** predefined dynamic variables has been duplicated and assigned a RADIUS tag value in the 100s range for use with this feature. The RADIUS tag value is the only difference between the existing CoS traffic-shaping predefined dynamic variables and the predefined dynamic variables that you must use with this feature.

Both RADIUS instances of the **$junos-cos-*parameter*** predefined dynamic variables are available, but you must use the dynamic variables with tag values in the 100s range to apply CoS traffic-shaping attributes to both the dynamic interface set and member subscriber sessions in a subscriber interface dynamic profile.

For example, the existing **$junos-cos-shaping-rate** predefined variable is assigned RADIUS vendor ID 4874, attribute number 108, and tag value 2. To apply CoS traffic-shaping attributes to the dynamic interface set and its member subscriber sessions, you must instead use the **$junos-cos-shaping-rate** predefined variable that is assigned RADIUS vendor ID 4874, attribute number 108, and tag value 102.

NOTE: Do not configure a combination of $junos-cos-*parameter* predefined dynamic variables with RADIUS tag values in the 100s range and $junos-cos-*parameter* predefined dynamic variables with tag values not in the 100s range in the same traffic-control profile. If you do so, the subscriber authentication process fails.

Related Documentation

- Guidelines for Configuring CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions on page 30
- Applying CoS Traffic-Shaping Attributes to Dynamic Interface Sets and Member Subscriber Sessions on page 71
- CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets on page 73

Guidelines for Configuring CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions

Observe the following guidelines when you apply dynamic CoS traffic-shaping attributes to a dynamic interface set or a dynamic ACI interface set and its member subscriber sessions. For complete information about the Junos OS CoS traffic-shaping predefined dynamic variables and RADIUS tag values used with this feature, see "CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets" on page 73.

- This feature is supported only for dynamically configured and instantiated subscriber interfaces.
- Do not configure a combination of **$junos-cos-*parameter*** predefined dynamic variables with RADIUS tag values in the 100s range and **$junos-cos-*parameter*** predefined dynamic variables with tag values not in the 100s range in the same traffic-control profile. If you do so, the subscriber authentication process fails.
- Use the **$junos-cos-adjust-minimum** predefined variable (tag 109) only in traffic-control profiles for dynamic subscriber interfaces. Using this variable in a traffic-control profile for a dynamic interface set or dynamic ACI interface set has no effect.
- Do not configure the **$junos-cos-excess-rate-high** predefined variable (tag 110) when the **$junos-cos-excess-rate** predefined variable (tag 105) is configured, and vice-versa.
- Do not configure the **$junos-cos-excess-rate-low** predefined variable (tag 111) when the **$junos-cos-excess-rate** predefined variable (tag 105) is configured, and vice-versa.
- Do not configure the **$junos-cos-byte-adjust-frame** predefined variable (tag 114) when the **$junos-cos-byte-adjust** predefined variable (tag 108) is configured, and vice-versa.
- Do not configure the **$junos-cos-byte-adjust-cell** predefined variable (tag 115) when the **$junos-cos-byte-adjust** predefined variable (tag 108) is configured, and vice-versa.
- Use the per-priority **$junos-cos-shaping-rate-*parameter*** predefined variables (tags 116 through 125) only in traffic-control profiles for dynamic interface sets or dynamic ACI interface sets. Using these variables in traffic-control profiles for a dynamic logical subscriber interface causes the subscriber session to fail.

Related Documentation

- Applying CoS Traffic-Shaping Attributes to Dynamic Interface Sets and Member Subscriber Sessions on page 71
- CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets on page 73
- CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions Overview on page 28

CHAPTER 4

CoS Hierarchical Scheduling on MPLS Ethernet Pseudowire Subscriber Interfaces

Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview

Junos OS supports two aspects of CoS for MPLS pseudowire subscriber interfaces. You can apply CoS rewrite rules and behavior aggregate (BA) classifiers to MPLS pseudowire subscriber interfaces. In addition, CoS performs egress hierarchical shaping towards the subscriber on MPLS pseudowire subscriber interfaces.

Hierarchical CoS enables you to apply traffic scheduling and queuing parameters and packet transmission scheduling parameters to an individual subscriber interface rather than to all interfaces configured on the port. Hierarchical CoS is supported on MX Series routers with either EQ DPCs or MPC/MICs installed.

On Juniper Networks MX Series routers, MPC/MIC and EQ DPC interfaces support a four-level CoS scheduling hierarchy that, when fully configured, consists of the physical interface (level 1), the interface set or the underlying interface (level 2), one or more logical interfaces (level 3), and one or more queues (level 4). Although all CoS scheduling hierarchies are four-level, level 1 is always the physical interface and level 4 is always the queue. Hierarchical scheduling configurations consist of the type of interfaces you configure; for example, a logical interface or an interface set and where those interfaces reside in the scheduling hierarchy, either level 2 or level 3. Because many hierarchical scheduling configurations are possible, we use the terms *two-level hierarchical scheduling* and *three-level hierarchical scheduling* in this discussion.

Related Documentation

- *Pseudowire Subscriber Logical Interfaces Overview*
- *Configuring a Pseudowire Subscriber Logical Interface*

- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- CoS Two-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 32
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 34
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 37
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

CoS Two-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces

Two-level hierarchical scheduling limits the number of hierarchical levels in the scheduling hierarchy to two. In a two-level scheduling hierarchy, all logical interfaces and interface sets share a single level 2 node.Table 11 on page 32 summarizes the interface hierarchy and the CoS scheduler node levels for two-level hierarchical scheduling.

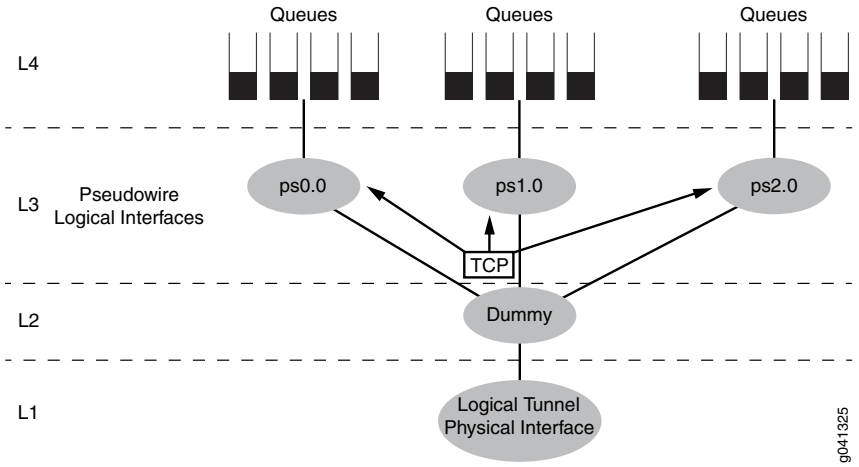
Table 11: Two-Level Hierarchical Scheduling-Interface Hierarchy Versus Scheduling Nodes

Level 1	Level 2	Level 3	Level 4
Physical interface	–	Pseudowire transport logical interface	One or more queues
Physical interface	–	Interface set	One or more queues
Physical interface	–	Pseudowire service logical interface	One or more queues

You use the two-level hierarchical scheduling when you have many pseudowires but you do not require shaping specific to the subscriber logical interface. For example, when your configuration is one subscriber per pseudowire interface.

Figure 8 on page 33 shows a two-level hierarchical scheduling configuration for the MPLS pseudowires. In this configuration, level 1 is the physical interface used for the logical tunnel anchor node. All of the pseudowire transport interfaces share a single level 2 node. The level 3 nodes are the pseudowire transport logical interfaces (ps0.0, ps1.0, and ps2.0). In this configuration, interface sets are not configured and only the logical interfaces have traffic control profiles.

Figure 8: MPLS Pseudowire Subscriber Interface Two-Level Scheduler Configuration



Two-level hierarchical scheduling has up to eight class of service queues. For this configuration, include the **maximum-hierarchy-levels 2** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement at the physical interface for the anchor logical tunnel.

NOTE: You cannot configure shaping policies on both the pseudowire logical interfaces and the subscriber logical interfaces over the same pseudowire. If a traffic-control profile is configured on a pseudowire logical interface, and CoS policies are configured on the subscriber logical interface over another pseudowire, all of the logical interfaces are at level 3 and act as peers.

Related Documentation

- *Pseudowire Subscriber Logical Interfaces Overview*
- *Configuring a Pseudowire Subscriber Logical Interface*
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 34
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 37
- Configuring CoS Two-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces on page 95
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces

In three-level hierarchical scheduling, the CoS scheduler nodes at level 1, level 2, and level 3 form a scheduling hierarchy. You can configure many different three-level scheduling hierarchies, depending on the location of the interface set and the use of underlying interfaces. In all variations, the physical interface on which the logical tunnel resides is a level 1 CoS scheduler node and the queues reside at level 4. Three-level scheduling hierarchies can have up to eight class of service queues.
Table 12 on page 34summarizes the most common three-level hierarchical scheduling configurations and shows the interface hierarchy and CoS scheduler nodes.

Table 12: Three-Level Hierarchical Scheduling-Interface Hierarchy Versus CoS Scheduling Node Levels

Level 1	Level 2	Level 3	Level 4
Physical interface	Pseudowire interface set	Pseudowire service logical interfaces	One or more queues
Physical interface	Pseudowire transport logical interface	Pseudowire interface set	One or more queues
Physical interface	Pseudowire transport logical interface	Pseudowire service logical interfaces	One or more queues

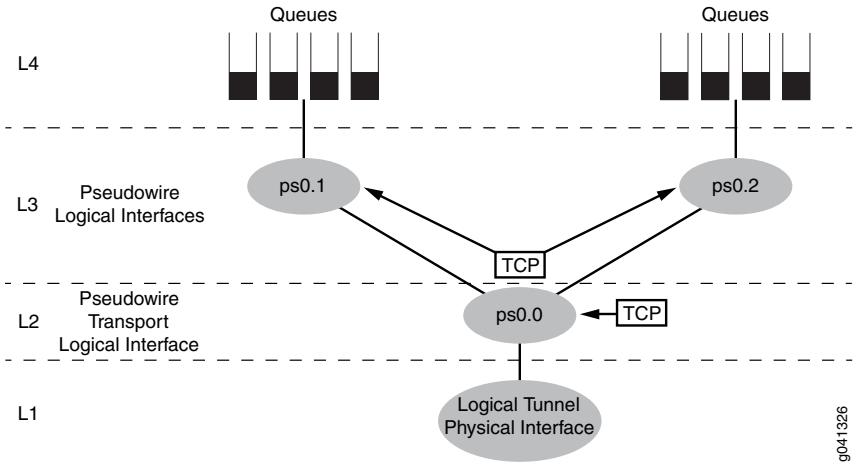
Three-Level Scheduling Hierarchy: Pseudowire Logical Interfaces over a Transport Logical Interface

Figure 9 on page 35 shows an MPLS pseudowire three-level scheduling hierarchy that includes two pseudowire service logical interfaces over a pseudowire transport logical interface. This variation uses the following scheduler nodes:

- Level 4—Forwarding class-based queues
- Level 3—Pseudowire service logical interfaces (ps0.1 and ps0.2) for subscriber sessions
- Level 2—Pseudowire transport logical interface (ps0.0)
- Level 1—Common/shared physical interface of the logical tunnel anchor point

You apply the traffic-control profiles at the pseudowire transport logical interfaces (level 2) and the pseudowire service logical interfaces (level 3).

Figure 9: Three-Level Scheduling Hierarchy Case 1: Pseudowire Service Logical Interfaces over a Transport Logical Interface



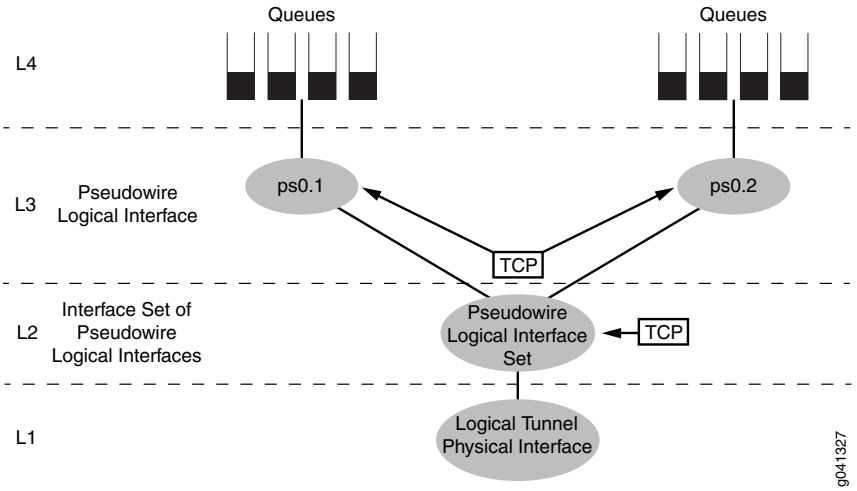
Three-Level Scheduling Hierarchy : Pseudowire Service Logical Interfaces over a Pseudowire Service Interface Set

Figure 10 on page 36 shows another variation of MPLS pseudowire three-level hierarchical scheduling that includes two pseudowire service logical interfaces over a pseudowire service interface set. This variation uses the following CoS scheduler nodes:

- Level 4—Forwarding class-based queues
- Level 3—Pseudowire service logical interfaces (ps0.1 and ps0.2)
- Level 2—Pseudowire service interface set
- Level 1—Common/shared physical interface of the logical tunnel anchor point

You apply the traffic-control profile at the pseudowire service interfaces (level 3) and at the interface set (level 2). This variation is most useful for subscriber edge deployments.

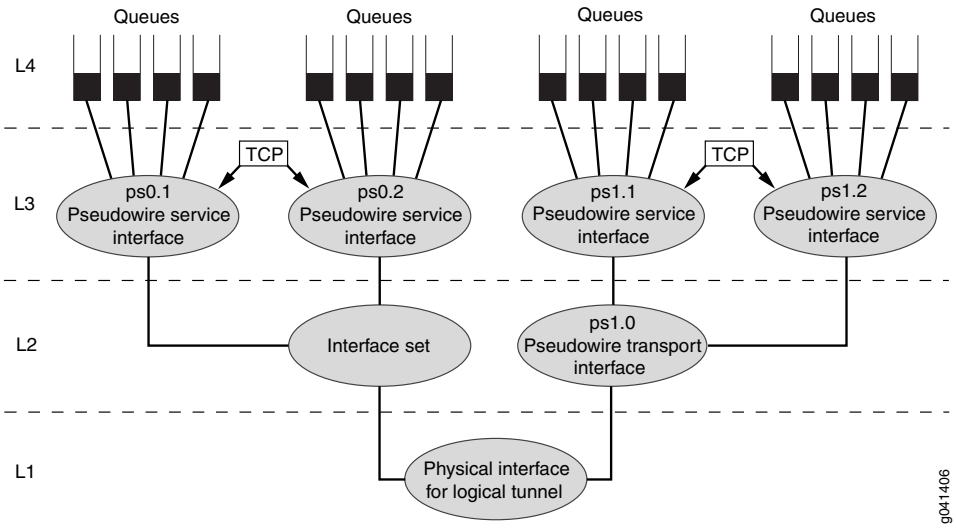
Figure 10: Three-Level Scheduling Hierarchy Case 2: Pseudowire Service Logical Interfaces over a Pseudowire Service Interface Set



Three-Level Scheduling Hierarchy Combined Deployment Scenario

Figure 11 on page 36 shows a deployment scenario that combines the three-level hierarchical scheduling scenarios in Figure 9 on page 35 and Figure 10 on page 36.

Figure 11: Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces-Deployment Scenario



This variation uses the following CoS scheduler nodes:

- Level 4—Forwarding class-based queues
- Level 3—Pseudowire service logical interfaces (ps0.1, ps0.2, ps1.1, and ps1.2)

- Level 2—Service interface set for pseudowire service interfaces (ps0.1 and ps0.2) and transport logical interface (ps1.0) for the pseudowire service logical interfaces (ps1.1 and ps1.2)
- Level 1—Common/shared physical interface of the logical tunnel anchor point

You apply the traffic-control profiles to the interfaces at both level 2 and level 3, as well as the interface set at level 2.

Related Documentation

- *Pseudowire Subscriber Logical Interfaces Overview*
- *Configuring a Pseudowire Subscriber Logical Interface*
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 37
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface) on page 97
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set) on page 99
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces

CoS supports two-level and three-level hierarchies for MPLS pseudowire subscriber interfaces.

To configure two-level scheduling, include the **maximum-hierarchy-levels 2** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement on the physical interface of the logical tunnel anchor point.

To configure three-level hierarchical scheduling, include the **implicit-hierarchy** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement on the physical interface of the logical tunnel anchor point. Use the following guidelines for configuring the **implicit-hierarchy** option:

- If an output traffic-control profile is configured on the pseudowire transport interface and on a pseudowire service interface, the two interfaces form a scheduling hierarchy. The pseudowire transport interface resides in a level 2 scheduler node and the pseudowire service interface resides in a level 3 scheduler node.
- If an output traffic-control profile is configured on the pseudowire services interface but not on a pseudowire transport interface, the pseudowire services interface resides in a level 3 scheduler node.
- If an output traffic-control profile is only configured on the pseudowire transport interface and not on the pseudowire services interface, the pseudowire transport interface resides in a level 3 scheduler node and all pseudowire traffic uses this node.

If the **implicit-hierarchy** option is not set on the logical tunnel anchor point, logical interfaces behave normally with the hierarchical-scheduler mode configured with or without the **hierarchical-scheduler maximum-hierarchy-levels** option under the **[edit interfaces *interface-name* hierarchical-scheduler]** statement. In this case, when you apply a traffic-control profile to the pseudowire and service logical interfaces, they both reside in level 3 scheduler nodes and do not form a scheduling hierarchy, which might not be the desirable behavior. In business edge, where only the pseudowire logical interfaces need to be shaped, applying the traffic-control profile at just the transport logical interface may be sufficient.

When configuring the logical tunnel physical interface for the maximum hierarchy level, all pseudowire logical interfaces operating on the physical interface use the same hierarchy model. If you want to mix two-level and three-level scheduling hierarchies, you can group the pseudowires together by hierarchy levels and share the same logical tunnel anchor point or you can use three-level scheduling for all pseudowires over the anchor point.

To specify rewrite rules and classifiers on pseudowire interfaces, reference the pseudowire device under the **[edit class-of-service interfaces]** hierarchy level and specify the rewrite rules and classifiers for the pseudowire interfaces.

To control all pseudowire traffic using the same logical tunnel interface, apply CoS policies at the physical interface for the anchor logical tunnel.

Related Documentation

- *Pseudowire Subscriber Logical Interfaces Overview*
- *Configuring a Pseudowire Subscriber Logical Interface*
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31
- Configuring CoS Two-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces on page 95
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface) on page 97
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set) on page 99
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

PART 2

Configuration

CHAPTER 5

Best Practices

Guidelines for Configuring Dynamic CoS for Subscriber Access

This topic describes the hardware requirements and guidelines for configuring dynamic CoS in a subscriber access environment.

Hardware Requirements for Dynamic CoS

Table 13 on page 42 lists the hardware requirements based on subscriber interface type for the hierarchical scheduling and per-unit scheduling dynamic CoS configurations.

Table 13: Hardware Required for Dynamic CoS Configurations

Dynamic CoS Configuration	Subscriber Interface Type	EQ DPCs on MX Series Routers	MPC/MIC Modules on MX Series Routers	IQ2 PICs on M120 and M320 Routers	IQ2E PICs on M120 and M320 Routers
Hierarchical CoS	Static and dynamic VLANs	Yes	Yes	No	No
	Static and dynamic VLANs over aggregated Ethernet	Yes	Yes	No	No
	Static or dynamic IP demux interfaces	Yes	Yes	No	No
	Static or dynamic IP demux interfaces over aggregated Ethernet	Yes	Yes	No	No
	Static or dynamic VLAN demux interfaces	No	Yes	No	No
	Static or dynamic VLAN demux interfaces over aggregated Ethernet	No	Yes	No	No
	Static PPPoE interfaces	No	Yes	Yes	Yes
	Dynamic PPPoE interfaces	No	Yes	No	Yes
	Static or dynamic PPPoE interfaces over aggregated Ethernet	No	Yes	No	No
	L2TP LAC tunnel over PPP	No	Yes	No	No
	L2TP LNS inline service over PPP	No	Yes	No	No

Table 13: Hardware Required for Dynamic CoS Configurations *(continued)*

Dynamic CoS Configuration	Subscriber Interface Type	EQ DPCs on MX Series Routers	MPC/MIC Modules on MX Series Routers	IQ2 PICs on M120 and M320 Routers	IQ2E PICs on M120 and M320 Routers
Per-unit scheduling	Static and dynamic VLANs	Yes	Yes	No	No
	Static and dynamic VLANs over aggregated Ethernet	No	No	No	No
	Static or dynamic IP demux interfaces	Yes	No	No	No
	Static or dynamic IP demux interfaces over aggregated Ethernet	No	No	No	No
	Static or dynamic VLAN demux interfaces	No	No	No	No
	Static or dynamic VLAN demux interfaces over aggregated Ethernet	No	No	No	No
	Static PPPoE interfaces	No	Yes	Yes	Yes
	Dynamic PPPoE interfaces	No	No	Yes	Yes
	Static or dynamic PPPoE interfaces over aggregated Ethernet	No	No	No	No
	L2TP LAC tunnel over PPP	No	No	No	No
	L2TP LNS inline service over PPP	No	No	No	No

Configuration Guidelines for Dynamic Scheduling and Queuing

When configuring scheduling and queuing for subscriber access, consider the following guidelines:

- To improve CoS performance in IPv4, IPv6, and dual-stack networks that use a DHCP access model, we recommend that you use the **aggregate-clients replace** statement rather than the **aggregate-clients merge** statement.
- You can configure dynamic CoS with one of the following scheduling configurations:
  - For hierarchical scheduling configurations, you must enable hierarchical scheduling in the static CLI for the interface referenced in the dynamic profile. If not, the dynamic profile fails.
  - For per-unit scheduling configurations, you must enable per-unit scheduling in the static CLI for the interface referenced in the dynamic profile. If not, the dynamic profile fails and schedulers are not attached to the interface.
- You configure the traffic scheduling and shaping parameters in a traffic-control profile within the dynamic profile. You can configure the scheduler map and schedulers in a dynamic profile or in the **[edit class-of-service]** hierarchy. You must statically configure the remaining CoS parameters, such as hierarchical scheduling, classifiers, drop profiles, and forwarding classes, in the **[edit class-of-service]** hierarchy.
- You can configure only one traffic-control-profile under a dynamic profile.
- You must define the output-traffic-control-profile that binds the traffic-control profile to the interface within the same dynamic profile as the interface.
- We recommend that you provide different names for the schedulers defined in dynamic profiles that are used for access and services. For example, if there are two dynamic profiles, voice-profile and video-profile, provide unique names for the schedulers defined under those profiles.
- You must use a service dynamic profile with a different profile name for each RADIUS CoA request over the same logical interface.
- When you configure scheduler and scheduler map sharing in client profiles, schedulers and scheduler maps must use the unique ID format. If the client profile uses the unique ID format and you want to have either scheduler or scheduler map sharing for service activation, you must configure the service profile in unique ID format.

Configuration Guidelines for Dynamic Classifiers and Rewrite Rules

When you configure classifiers and rewrite rules for subscriber access, consider the following guidelines:

- To apply classifiers and rewrite rules to a subscriber interface in a dynamic profile, you must configure the rewrite rule and classifier definitions in the static **[edit class-of-service]** hierarchy and reference them in the dynamic profile.

- If a static classifier or a rewrite rule definition that is referenced by a dynamic subscriber interface does not exist, the configuration is invalid and the subscriber cannot log in.
- If a network administrator changes the static classifiers and rewrite rules definitions that are referenced in a dynamic profile with an active subscriber interface logged in, the changes are applied to the active subscriber interface immediately.
- If a network administrator deletes a classifier or a rewrite rule definition that is referenced by an active dynamic subscriber interface, the system removes the classifier or rewrite rule binding from the interface. The classifier is replaced by the default classifier. If the network administrator adds the removed classifier or rewrite rule to the configuration while the dynamic interface is active, the addition does not take effect until the subscriber logs out and then logs in again.

- IP demux interfaces can only instantiate Layer 3 rules (both rewrite rules and classifiers).
  - An IP demux subscriber interface can implicitly inherit a classifier from the underlying interface. If an IP demux interface is created without a classifier and a Layer 2 classifier is attached to the underlying interface, the IP demux interface also inherits the Layer 2 classifier. The show class-of-service interface ***interface-name*** command does not display this attachment.

    Table 14 on page 45 lists the classification rule configuration for an IP demux subscriber interface with a VLAN underlying interface.

Table 14: IP Demux Classification Rules

VLAN Underlying Interface Classifier Configuration	IP Demux Interface Classifier Configuration	Resulting Classifier Configuration
Layer 2	—	VLAN Layer 2
Layer 2	Layer 3	Demux Layer 3
Layer 3	—	Default
Layer 3	Layer 3	Demux Layer 3

  - An IP demux subscriber interface explicitly inherits Layer 2 rewrite rules from the underlying interface if a Layer 2 rewrite rule is present. The show class-of-service interface ***interface-name*** command displays the attachment.

    Table 15 on page 45 lists the rewrite rule configuration for an IP demux subscriber interface with a VLAN underlying interface.

Table 15: IP Demux Rewrite Rules

VLAN Underlying Interface Rewrite Rule Configuration	IP Demux Interface Rewrite Rule Configuration	Resulting Rewrite Rule Configuration
Layer 2	—	VLAN Layer 2

Table 15: IP Demux Rewrite Rules *(continued)*

VLAN Underlying Interface Rewrite Rule Configuration	IP Demux Interface Rewrite Rule Configuration	Resulting Rewrite Rule Configuration
Layer 2	Layer 3	VLAN Layer 2 and demux Layer 3
Layer 3	—	Default
Layer 3	Layer 3	Demux Layer 3

- An L2TP subscriber interface can implicitly inherit a classifier from the underlying interface.

  Table 16 on page 46 lists the classification rule configuration for an L2TP LAC subscriber interface with a VLAN underlying interface.

Table 16: L2TP Classification Rules

VLAN Underlying Interface Classifier Configuration	L2TP LAC Classifier Configuration	Resulting Classifier Configuration
Layer 2 or Fixed	Layer 2 or Fixed	VLAN Layer 2 or Fixed
Layer 2 or Fixed	Layer 3	Demux/PPPoE Layer 3
Layer 3	Layer 2 or Fixed	VLAN Layer 2 or Fixed
Layer 3	Layer 3	Demux/PPPoE Layer 3

- An L2TP LAC subscriber interface explicitly inherits Layer 2 rewrite rules from the underlying interface if a Layer 2 rewrite rule is present. Table 17 on page 46 lists the rewrite rule configuration for an L2TP LAC subscriber interface with a VLAN underlying interface.

Table 17: L2TP LAC Rewrite Rules

VLAN Underlying Interface Rewrite Rule Configuration	L2TP Interface Rewrite Rule Configuration	Resulting Rewrite Rule Configuration
Layer 2	Layer 2	VLAN Layer 2
Layer 2	Layer 3	VLAN Layer 2 and demux/PPPoE Layer 3
Layer 3	Layer 2	VLAN Layer 2 and demux/PPPoE Layer 3
Layer 3	Layer 3	Demux/PPPoE Layer 3

Configuration Guidelines for Dynamic CoS on Specific Interface Types

To obtain configuration guidelines for CoS on specific interface types, see:

- CoS for Aggregated Ethernet Subscriber Interfaces Overview on page 11
- CoS for L2TP LAC Subscriber Interfaces Overview on page 13
- CoS for L2TP LNS Inline Services Overview on page 15
- CoS for PPPoE Subscriber Interfaces Overview on page 12
- CoS for Interface Sets of Subscribers Overview on page 17

Related Documentation

- CoS for Subscriber Access Overview on page 3
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- Configuring Per-Unit Scheduling in a Dynamic Profile for Subscriber Access on page 53
- *Configuring Static CoS for an L2TP LNS Inline Service*

CHAPTER 6

Configuration Overview

Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access

You configure static scheduling and queuing in a dynamic profile for subscriber access.

To configure CoS in a dynamic profile for subscriber access using static scheduling and queuing parameters:

1. Configure the static CoS parameters in the **[edit class-of-service]** hierarchy.
   a. Enable the hierarchical scheduler for the interface.

      See "Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers" on page 4.

   b. Configure the scheduler map and schedulers.

      When you configure static scheduling and queuing in a dynamic profile, you reference the scheduler map in the dynamic profile.

      See *Configuring Schedulers*.

   c. Configure the drop profiles.

      See *Configuring RED Drop Profiles*.

   d. Configure the forwarding classes.

      See *Configuring Forwarding Classes*.

   e. Configure the rewrite-rules and classifier definitions.

      See *Configuring Rewrite Rules* and *Defining Classifiers*.

See *Junos CoS Components* for information about configuring the remaining CoS parameters.

2. Configure a static or dynamic subscriber interface that can be referenced in the dynamic profile.
   - For static VLAN interfaces, see *Configuring Static Subscriber Interfaces in Dynamic Profiles*.
   - For dynamic VLAN interfaces, see *Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet*.
   - For dynamic IP demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles* and *Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet*.
   - For dynamic VLAN demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles*.
   - For dynamic PPPoE interfaces, see *Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles*.
3. Configure CoS parameters in a dynamic profile.
   a. Configure the dynamic profile.

      See *Configuring a Basic Dynamic Profile*.

   b. Configure traffic shaping and scheduling parameters in the dynamic profile using a traffic-control profile.

      Reference the scheduler map you configured in the static **[edit class-of-service]** hierarchy.

      See "Configuring Static Traffic Shaping and Scheduling Parameters in a Dynamic Profile" on page 57.

   c. Apply CoS parameters to a subscriber interface by referencing an interface in the dynamic profile.

      See "Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile" on page 81.

4. To configure default values for subscribers on login, and enable subscribers to replace other CoS parameters when replacing services, configure variables in the dynamic profile.

   See "Configuring User-Defined CoS Variables in a Dynamic Service Profile" on page 68.

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for Subscriber Access Overview on page 3
- Example: Configuring Static Hierarchical Scheduling and Queuing for Subscriber Access on page 101

Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access

You can configure dynamic scheduling and queuing in dynamic profile for subscriber access.

To configure dynamic scheduling and queuing for subscriber access using dynamic scheduling and queuing parameters:

1. Configure the static CoS parameters in the **[edit class-of-service]** hierarchy.
   a. Enable the hierarchical scheduler for the interface.

      See "Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers" on page 4 and **hierarchical-scheduler**.

   b. Configure the drop profiles.

      See *Configuring RED Drop Profiles*.

   c. Configure the forwarding classes.

      See *Configuring Forwarding Classes*.

   d. Configure the rewrite-rules and classifier definitions.

      See *Configuring Rewrite Rules* and *Defining Classifiers*.

   See *Junos CoS Components* for information about configuring the remaining CoS parameters.
2. Configure a static or dynamic subscriber interface that can be referenced in the dynamic profile.
   - For static VLAN interfaces, see *Configuring Static Subscriber Interfaces in Dynamic Profiles*.
   - For dynamic VLAN interfaces, see *Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet*.
   - For dynamic IP demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles* and *Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet*.
   - For dynamic VLAN demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles*.
   - For dynamic PPPoE interfaces, see *Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles*.

3. Configure CoS parameters in a dynamic profile.

   a. Configure the dynamic profile.

      See *Configuring a Basic Dynamic Profile*.

   b. Configure traffic shaping and scheduling parameters in the dynamic profile using a traffic-control profile.

      See "Configuring Traffic Scheduling and Shaping for Subscriber Access" on page 57.

   c. Configure the schedulers and scheduler map in the dynamic profile.

      You can configure the schedulers using dynamic variables or a combination of both static values and dynamic variables.

      See "Configuring Schedulers in a Dynamic Profile for Subscriber Access" on page 59.

   d. Apply CoS parameters to a subscriber interface by referencing an interface in the dynamic profile.

      - For traffic shaping and scheduling, see "Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile" on page 81.
      - For rewrite-rules, see "Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile" on page 83.
      - For classifiers, see "Applying a Classifier to a Subscriber Interface in a Dynamic Profile" on page 84.

4. (Optional) Configure variables in access and service profiles to enable RADIUS to activate subscriber and upgrade services through CoA.

   NOTE: **Do not instantiate a CoA request using a service dynamic profile that is already in use on the same logical interface.**

   a. Configure user-defined CoS variables in a dynamic profile.

      See "Configuring User-Defined CoS Variables in a Dynamic Service Profile" on page 68

   b. (Optional) Enable multiple clients for the same subscriber (logical interface) to aggregate attributes by configuring the **aggregate-clients** option for the dynamic profile attached to a DHCP subscriber interface.

      See *Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces*.

      Because you have configured the scheduler map in the dynamic profile, queues are merged when subscribers change services. Other CoS parameters are replaced.

      When multiple subscribers are enabled on a DHCP subscriber interface, and the dynamic profile referenced by DHCP does not have the **replace** keyword configured, the system does not replace the parameters. Instead, it combines the values of the parameters to their maximum scalar value.

Related Documentation

- hierarchical-scheduler on page 168
- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for Subscriber Access Overview on page 3
- Example: Configuring Dynamic Hierarchical Scheduling and Queuing for Subscriber Access on page 103

Configuring Per-Unit Scheduling in a Dynamic Profile for Subscriber Access

Per-unit scheduling enables one set of output queues for each logical interface configured under the physical interface. In per-unit scheduling configurations, each Layer 3 scheduler node is allocated a dedicated set of queues.

If you do not explicitly configure CoS parameters, a default traffic profile with queues is attached to the logical interface.

To configure per-unit scheduling and queuing for subscriber access:

1. Configure the static CoS parameters in the **[edit class-of-service]** hierarchy.
   a. Enable the per-unit scheduler for the physical interface.

      [edit interfaces *interface-name*]
      user@host# **set per-unit-scheduler**

   b. Configure the drop profiles.

      See *Configuring RED Drop Profiles*.

   c. Configure the forwarding classes.

      See *Configuring Forwarding Classes*.

   d. Configure the rewrite-rules and classifier definitions.

      See *Configuring Rewrite Rules* and *Defining Classifiers*.

   See *Junos CoS Components* for information about configuring the remaining CoS parameters.
2. Configure a static or dynamic subscriber interface that can be referenced in the dynamic profile.
   - For static VLAN interfaces, see *Configuring Static Subscriber Interfaces in Dynamic Profiles*.
   - For dynamic IP demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles*.
   - For dynamic PPPoE interfaces, see *Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles*.

3. Configure CoS parameters in a dynamic profile.

   a. Configure the dynamic profile.

      See *Configuring a Basic Dynamic Profile*.

   b. Configure traffic shaping and scheduling parameters in the dynamic profile using a traffic-control profile.

      See "Configuring Traffic Scheduling and Shaping for Subscriber Access" on page 57.

   c. Configure the schedulers and scheduler map in the dynamic profile.

      You can configure the schedulers using dynamic variables or a combination of both static values and dynamic variables.

      See "Configuring Schedulers in a Dynamic Profile for Subscriber Access" on page 59.

   d. Apply CoS parameters to a subscriber interface by referencing an interface in the dynamic profile.

      - For traffic shaping and scheduling, see "Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile" on page 81.
      - For rewrite rules, see "Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile" on page 83.
      - For classifiers, see "Applying a Classifier to a Subscriber Interface in a Dynamic Profile" on page 84.

4. (Optional) Configure variables in access and service profiles to enable RADIUS to activate subscriber and upgrade services through CoA.

   NOTE: **Do not instantiate a CoA request using a service dynamic profile that is already in use on the same logical interface.**

   Because you have configured the scheduler map in the dynamic profile, queues are merged when subscribers change services. Other CoS parameters are replaced.

   When multiple subscribers are enabled on a DHCP subscriber interface, and the dynamic profile referenced by DHCP does not have the **replace** keyword configured, the system does not replace the parameters. Instead, it combines the values of the parameters to their maximum scalar value.

   a. Configure CoS variables in a dynamic profile.

      See "Configuring User-Defined CoS Variables in a Dynamic Service Profile" on page 68

   b. (Optional) Enable multiple clients for the same subscriber (logical interface) to aggregate attributes by configuring the **aggregate-clients** option for the dynamic profile attached to a DHCP subscriber interface.

      See *Attaching Dynamic Profiles to DHCP Subscriber Interfaces or DHCP Client Interfaces*.

Related Documentation

- CoS for Subscriber Access Overview on page 3
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Example: Configuring Per-Unit Scheduling for Subscriber Access on page 110

CHAPTER 7

Configuration Tasks for Scheduling and Queuing

Configuring Traffic Scheduling and Shaping for Subscriber Access

You use traffic-control profiles to configure traffic shaping and scheduling properties. When you reference a traffic-control profile in a dynamic profile, you can provide hierarchical shaping and scheduling for a subscriber interface.

You can choose to configure static values or dynamic variables for the shaping parameters. The values for the dynamic variables are obtained from RADIUS when a subscriber logs in or when a subscriber changes services.

You cannot configure a traffic-control profile that contains a combination of static and dynamic parameters.

This topic includes the following tasks:

Configuring Static Traffic Shaping and Scheduling Parameters in a Dynamic Profile

To configure static traffic shaping and scheduling parameters in a traffic-control profile:

1. Create the traffic-control profile and assign a name.

   [edit dynamic-profiles business-profile class-of-service]
   user@host# edit traffic-control-profiles ***profile-name***

2. Do one of the following:

   - Reference a static scheduler map in the dynamic profile. The scheduler map is statically configured in the **[edit class-of-service]** hierarchy.

[edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
user@host# set scheduler-map *map-name*

- Reference a dynamic scheduler map in the dynamic profile. The scheduler map is dynamically configured in the **[edit dynamic-profiles *profile-name* class-of-service scheduler-maps]** hierarchy.

  [edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
  user@host# set scheduler-map *map-name*

3. Configure the shaping rate to be used in the dynamic profile.

   [edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
   user@host# set shaping-rate (*rate* <burst-size *bytes*>

4. Configure the guaranteed rate to be used in the dynamic profile.

   [edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
   user@host# set guaranteed-rate (*rate* <burst-size *bytes*>

5. Configure the delay-buffer rate.

   If you do not include this statement, the delay-buffer rate is based on the guaranteed rate if one is configured, or on the shaping rate if no guaranteed rate is configured.

   [edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
   user@host# set delay-buffer-rate (percent *percentage* | *rate*)

Configuring Dynamic Traffic Shaping and Scheduling Parameters in a Dynamic Profile

You can configure variables for the traffic shaping and scheduling parameters. The values for the parameters are dynamically obtained by RADIUS when a subscriber logs in or changes a service.

To configure dynamic traffic-control profiles in a dynamic profile:

1. Create the traffic-control profile.

   [edit dynamic-profiles business-profile class-of-service]
   user@host# edit traffic-control-profiles *profile-name*

2. Configure the scheduler map variable.

   [edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
   user@host# set scheduler-map $junos-cos-scheduler-map

3. Configure the shaping rate variable.

   [edit dynamic-profiles business-profile class-of-service traffic-control-profiles *profile-name*]
   user@host# set shaping-rate $junos-cos-shaping-rate <burst-size *bytes*>

4. Configure the guaranteed rate variable.

```
[edit dynamic-profiles business-profile class-of-service traffic-control-profiles
  profile-name]
user@host# set guaranteed-rate $junos-cos-guaranteed-rate <burst-size [ bytes |
  $junos-cos-guaranteed-rate-burst]>
```

5. Configure a variable for the delay-buffer rate.

   If you do not include this statement, the delay-buffer rate is based on the guaranteed rate if one is configured, or the shaping rate if no guaranteed rate is configured.

   ```
   [edit dynamic-profiles business-profile class-of-service traffic-control-profiles
     profile-name]
   user@host# set delay-buffer-rate $junos-cos-delay-buffer-rate
   ```

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for Subscriber Access Overview on page 3
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- Example: Configuring Static Hierarchical Scheduling and Queuing for Subscriber Access on page 101
- Example: Configuring Dynamic Hierarchical Scheduling and Queuing for Subscriber Access on page 103
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195

Configuring Schedulers in a Dynamic Profile for Subscriber Access

You use schedulers to define the parameters of output queues. These properties include the amount of interface bandwidth assigned to the queue, the size of the memory buffer allocated for storing packets, the priority of the queue, and the tail drop profiles associated with the queue.

You can configure up to four schedulers in a dynamic profile.

Within a dynamic profile, you can choose to define schedulers with static values, dynamic variables, or a combination of static values and dynamic variables. The dynamic variables enable RADIUS to provide the value for the scheduler parameter when the subscriber logs in.

- Configuring Static Schedulers in a Dynamic Profile on page 60
- Configuring Dynamic Schedulers with Variables in a Dynamic Profile on page 61
- Configuring a Combination of Static and Dynamic Scheduler Parameters in a Scheduler Definition on page 62

Configuring Static Schedulers in a Dynamic Profile

This topic describes how to configure schedulers with static values in a dynamic profile for subscriber access.

To configure static scheduling and queuing in a dynamic profile:

1. Configure the scheduler and queuing parameters.

   a. Specify the scheduler for which you want to configure parameters.

      [edit dynamic-profiles *profile-name* class-of-service]
      user@host# **edit** schedulers ***scheduler-name***

   b. Configure the buffer size.

      [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
      user@host# **set** buffer-size **remainder**

   c. Configure the drop-profile map and drop profile.

      [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
      user@host# **set** drop-profile-map **loss-priority any protocol any** drop-profile **d3**

   d. Configure the priority.

      [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
      user@host# **set** priority **low**

   e. Configure the transmit rate.

      [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
      user@host# **set** transmit-rate **percent 40**

   f. Configure the excess rate.

      [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
      user@host# **set excess-rate percent 90**

   g. (Optional) Configure the priority value for the excess-rate.

      [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
      user@host# **set excess-priority high**

2. Associate the scheduler with a scheduler map.

   a. Configure the scheduler map name.

      [edit dynamic-profiles *profile-name* class-of-service]
      user@host# **set** scheduler-maps **data-smap**

   b. Configure the forwarding class.

      [edit dynamic-profiles *profile-name* class-of-service scheduler-maps *map-name*]
      user@host# **set** forwarding-class **be**

   c. Configure the scheduler.

      [edit dynamic-profiles *profile-name* class-of-service scheduler-maps *map-name*
      forwarding-class *forwarding-class-name*]

```
user@host# set scheduler be_sch
```

Configuring Dynamic Schedulers with Variables in a Dynamic Profile

You can configure variables for the dynamic scheduler parameters. These values are dynamically obtained by RADIUS when a subscriber logs in or changes a service using a RADIUS change of authorization (CoA) message.

To configure dynamic scheduling and queuing in a dynamic profile:

1. Configure the scheduler and queuing parameters.
   a. Specify the scheduler name using a variable.

      ```
      [edit dynamic-profiles profile-name class-of-service]
      user@host# edit schedulers $junos-cos-scheduler
      ```

   b. Configure the variable for the buffer size.

      ```
      [edit dynamic-profiles profile-name class-of-service schedulers]
      user@host# set buffer-size (percent $junos-cos-scheduler-bs | temporal
        $junos-cos-scheduler-bs)
      ```

   c. Configure the variables for the drop-profile maps and the drop profile.

      ```
      [edit dynamic-profiles profile-name class-of-service schedulers]
      user@host# set drop-profile-map loss-priority low protocol any drop-profile
        $junos-cos-scheduler-low
      user@host# set drop-profile-map loss-priority medium-low protocol any
        drop-profile $junos-cos-scheduler-medium-low
      user@host# set drop-profile-map loss-priority medium-high protocol any
        drop-profile $junos-cos-scheduler-medium-high
      user@host# set drop-profile-map loss-priority high protocol any drop-profile
        $junos-cos-scheduler-high
      user@host# set drop-profile-map loss-priority any protocol any drop-profile
        $junos-cos-scheduler-any
      ```

   d. Configure the variable for the priority.

      ```
      [edit dynamic-profiles profile-name class-of-service schedulers]
      user@host# set priority $junos-cos-scheduler-pri
      ```

   e. Configure the variable for the transmit rate.

      ```
      [edit dynamic-profiles profile-name class-of-service schedulers]
      user@host# set transmit-rate $junos-cos-scheduler-tx
      ```

   f. Configure the variable for the excess rate.

      ```
      [edit dynamic-profiles profile-name class-of-service schedulers]
      user@host# set excess-rate percent $junos-cos-scheduler-excess-rate
      ```

   g. Configure the variable for the priority of the excess-rate.

      ```
      [edit dynamic-profiles profile-name class-of-service schedulers]
      user@host# set excess-priority $junos-cos-scheduler-excess-priority
      ```

2. Associate the scheduler with a scheduler map.

   a. Configure the scheduler map name.

      [edit dynamic-profiles *profile-name* class-of-service]
      user@host# **edit** scheduler-maps ***scheduler-map-name***

   b. Configure the forwarding class.

      [edit dynamic-profiles *profile-name* class-of-service scheduler-maps
      scheduler-map-name]
      user@host# **set** forwarding-class **be**

   c. Configure the scheduler.

      [edit dynamic-profiles *profile-name* class-of-service scheduler-maps
      scheduler-map-name]
      user@host# **set** scheduler **$junos-cos-scheduler**

Configuring a Combination of Static and Dynamic Scheduler Parameters in a Scheduler Definition

Within a dynamic profile, you can choose to configure one dynamic scheduler definition, or combine static and dynamic scheduler parameters in many static scheduler definitions.

Combining static and dynamic scheduler parameters enables you to provide subscribers with unique rate configurations that the RADIUS definitions for predefined variables do not allow.

To configure a scheduler definition that contains static and dynamic scheduling and queuing parameters:

1. Configure the scheduler definition.

   a. Specify the scheduler name.

      NOTE: **To configure a static scheduler that contains both static and dynamic parameters, you must specify a unique scheduler name, not the $junos-cos-scheduler variable.**

      [edit dynamic-profiles *profile-name* class-of-service]
      user@host# **edit** schedulers ***scheduler-name***

   b. Configure the buffer size.

      Do either of the following:

      - Configure a static value.

        [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
        user@host# **set** buffer-size **(percent *percentage* | remainder | temporal (*microseconds*)**

      - Configure a variable.

        [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]

```
user@host# set buffer-size (percent $junos-cos-scheduler-bs | temporal
  $junos-cos-scheduler-bs)
```

c. Configure the drop-profile maps, the drop profile, and the priority.

   Do either of the following:

   - Configure static values.

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set drop-profile-map loss-priority any protocol any drop-profile d3
     ```

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set priority low
     ```

   - Configure variables.

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set drop-profile-map loss-priority low protocol any drop-profile
       $junos-cos-scheduler-low
     user@host# set drop-profile-map loss-priority medium-low protocol any
       drop-profile $junos-cos-scheduler-medium-low
     user@host# set drop-profile-map loss-priority medium-high protocol any
       drop-profile $junos-cos-scheduler-medium-high
     user@host# set drop-profile-map loss-priority high protocol any drop-profile
       $junos-cos-scheduler-high
     user@host# set drop-profile-map loss-priority any protocol any drop-profile
       $junos-cos-scheduler-any
     ```

d. Configure the priority.

   Do either of the following:

   - Configure a static value.

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set excess-priority high
     ```

   - Configure a variable.

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set excess-priority $junos-cos-scheduler-excess-priority
     ```

e. Configure the transmit rate.

   Do either of the following:

   - Configure a static value.

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set transmit-rate
     ```

   - Configure a variable.

     ```
     [edit dynamic-profiles profile-name class-of-service schedulers scheduler-name]
     user@host# set transmit-rate $junos-cos-scheduler-tx
     ```

f. Configure the excess rate.

Do either of the following:

- Configure a static value.

  [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
  user@host# **set excess-rate percent 250**

- Configure a variable.

  [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
  user@host# **set excess-rate percent $junos-cos-scheduler-excess-rate**

g. Configure the priority for the excess-rate.

Do either of the following:

- Configure a static value.

  [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
  user@host# **set excess-priority high**

- Configure a variable.

  [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]
  user@host# **set excess-priority percent $junos-cos-scheduler-excess-priority**

2. Associate the scheduler with a scheduler map.

a. Configure the scheduler map name.

[edit dynamic-profiles *profile-name* class-of-service]
user@host# **edit** scheduler-maps ***scheduler-map-name***

b. Configure the forwarding class.

[edit dynamic-profiles *profile-name* class-of-service scheduler-maps *scheduler-map-name*]
user@host# **set** forwarding-class **be**

c. Configure the scheduler.

[edit dynamic-profiles *profile-name* class-of-service scheduler-maps *scheduler-map-name*]
user@host# **set** scheduler **$junos-cos-scheduler**

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195
- Changing CoS Services Overview on page 25

Configuring Scheduler and Scheduler Map Sharing

The system generates unique identifiers (IDs) in dynamic profiles created for services. The generated unique IDs enable you to identify and configure separate parameter values with the same variable name. When applied to CoS, you can now configure scheduler and scheduler map sharing. In client-access profiles, schedulers and scheduler maps must use the unique ID format. If the client-access profile uses the unique ID format and you want to have either scheduler or scheduler map sharing for service activation, you must configure the service profile in unique ID format. Generating unique IDs based on schedulers and scheduler maps eliminates duplication and improves router performance and scalability. You can configure scheduler and scheduler map sharing by including the variables for CoS in the client access or service dynamic profile. All scheduler maps and schedulers must be in the unique ID format.

Before you configure variables for the client access or service dynamic profile:

- Create a basic dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

To configure variables for the client access or service dynamic profile:

1. Configure the variables for the dynamic client access profile.

   [edit dynamic-profiles *client-profile* variables]
   user@host# **set smap_data uid**
   user@host# **set data_sched uid**

2. Configure the CoS parameters for the variables in the scheduler profile.

   [edit dynamic-profiles*client-profile* class-of-service]
   user@host# **edit schedulers "$data_sched"**
   user@host# **set transmit-rate percent 10**
   user@host# **set buffer-size remainder**
   user@host# **set priority low**

3. Configure the CoS parameters for the variables in the scheduler maps profile.

   [edit dynamic-profiles *client-profile* class-of-service]
   user@host# **edit scheduler-maps "$smap_data"**
   user@host# **edit forwarding-class be scheduler "$data_sched"**

For example, you can configure scheduler maps and schedulers for a client access profile:

```
dynamic-profiles {
  cos-para {
    variables {
      data_smap uid;
      data_video_smap uid;
      voice_smap uid;
      data_sched uid;
      video_sched uid;
      voice_sched uid;
    }
    ...
```

```
class-of-service {
  traffic-control-profiles {
    tcp1 {
      scheduler-map "$junos-cos-scheduler-map";
      shaping-rate "$junos-cos-shaping-rate";
      guaranteed-rate 10m;
      delay-buffer-rate "$junos-cos-delay-buffer-rate";
    }
  }
  interfaces {
    "$junos-interface-ifd-name" {
      unit "$junos-underlying-interface-unit" {
        output-traffic-control-profile tcp1;
      }
    }
  }
  scheduler-maps {
    "$data_smap" {
      forwarding-class be scheduler "$data_sched";
    }
    "$data_video_smap" {
      forwarding-class be scheduler "$data_sched";
      forwarding-class af scheduler "$video_sched";
    }
    “$voice_smap” {
      forwarding-class ef scheduler “$voice_sched”;
    }
  }
  schedulers {
    "$data_sched" {
      transmit-rate "$junos-cos-scheduler-tx";
      inactive: buffer-size percent "$junos-cos-scheduler-bs";
      priority "$junos-cos-scheduler-pri";
    }
    "$video_sched" {
      transmit-rate "$junos-cos-scheduler-tx";
      inactive: buffer-size percent "$junos-cos-scheduler-bs";
      priority "$junos-cos-scheduler-pri";
    }
    “$voice_sched” {
      transmit-rate percent 10;
      buffer-size remainder;;
      priority low;
    }
  }
 }
}
}
```

Related Documentation

- *Access Profiles and Service Profiles Overview*
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41

CHAPTER 8

Configuration Tasks for Applying CoS Using RADIUS

Configuring Initial CoS Parameters Dynamically Obtained from RADIUS

You can configure a subscriber interface so that subscribers receive initial CoS parameters that the router obtains from the RADIUS authentication server when subscribers log in using that logical interface on the router.

1. Configure external RADIUS server VSAs with values that you expect subscribers to log in with.
   - To configure a RADIUS authentication server to include CoS traffic-shaping parameters in authentication grants on certain subscriber interfaces, configure Juniper Networks VSA 26–108.
   - To configure a RADIUS authentication server to include CoS scheduling and queuing parameters in authentication grants a certain subscriber interfaces, configure Juniper Networks VSA 28–146.

   See *Configuring Router or Switch Interaction with RADIUS Servers* and *Configuring RADIUS Server Parameters for Subscriber Access*.
2. Configure a subscriber interface that supports hierarchical CoS.
   - For static VLAN interfaces, see *Configuring Static Subscriber Interfaces in Dynamic Profiles*.
   - For static VLAN interfaces over aggregated Ethernet, see *Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet*.

- For static IP demux interface sets, see *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles*.
- For dynamic IP demux interface sets, see *Configuring a Subscriber Interface Using a Set of Static IP Demux Interfaces*

3. Associate a traffic-control profile with the interface.

   See "Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile" on page 81.

4. Configuring initial traffic-shaping parameters to be obtained from RADIUS.

   See "Configuring Dynamic Traffic Shaping and Scheduling Parameters in a Dynamic Profile" on page 58.

5. Configure forwarding classes and scheduler maps statically.

   See *Configuring Forwarding Classes* and *Configuring Scheduler Maps*.

6. Configure a scheduler to specify initial scheduling and queuing parameters to be dynamically obtained from RADIUS when a subscriber logs in.

   See "Configuring Dynamic Schedulers with Variables in a Dynamic Profile" on page 61.

Related Documentation

- Subscriber Interfaces That Provide Initial CoS Parameters Dynamically Obtained from RADIUS on page 21
- Example: Configuring Initial CoS Parameters Dynamically Obtained from RADIUS on page 119
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- *Subscriber Activation and Service Management in an Access Network*
- *Juniper Networks VSAs Supported by the AAA Service Framework*
- *Dynamic Profiles Overview*
- *Dynamic Variables Overview*
- *Junos OS Predefined Variables*

Configuring User-Defined CoS Variables in a Dynamic Service Profile

You can configure user-defined variables in the dynamic service profile for traffic scheduling and shaping parameters.

You can use variables in a dynamic service profile in two ways:

- To enable subscribers to upgrade or downgrade services after login using a RADIUS change of authorization (CoA), configure user-defined variables for CoS parameters as RADIUS attributes.
- To provide subscribers with default values for CoS parameters, configure user-defined variables for CoS parameters with static default values. If you have configured values

to be supplied by a RADIUS CoA, subscribers can receive the previously configured default value when deactivating a service.

You activate the variables by referencing them in the traffic-control profile configured in the dynamic service profile.

To configure user-defined variables for CoS in a dynamic profile:

1. Specify that you want to configure variables in the dynamic profile.

   [edit dynamic-profiles residential-silver variables]

2. Do one of the following to configure variables for the shaping rate:

   - Enable RADIUS to modify the shaping rate based on service changes.

     a. Configure the attribute:

        [edit dynamic-profiles residential-silver variables]
        user@host# set srate radius vendor-id 4874 attribute 108

     b. Configure the tag:

        [edit dynamic-profiles residential-silver variables]
        user@host# set srate radius vendor-id 4874 tag 2

     NOTE: You can configure user-defined values for RADIUS tags that are different than the values that are required in access profiles with predefined variables. For example, in a dynamic service profile, you can assign the shaping rate with a tag of 1 rather than 2, which is required for the **$junos-shaping-rate** variable. When you configure user-defined values, the VSA that is sent from RADIUS must share the same definition.

   - Configure a default value for the shaping rate.

     [edit dynamic-profiles residential-silver variables]
     user@host# set srate default-value 10m

3. Do one of the following to configure variables for the guaranteed rate:

   - Enable RADIUS to modify the guaranteed rate based on service changes.

     a. Configure the attribute.

        [edit dynamic-profiles residential-silver variables]
        user@host# set grate radius vendor-id 4874 attribute 108

     b. Configure the tag.

        [edit dynamic-profiles residential-silver variables]
        user@host# set grate radius vendor-id 4874 tag 3

   - Configure a default value for the guaranteed rate.

     [edit dynamic-profiles residential-silver variables]
     user@host# set grate default-value 5m

4. Do one of the following to configure variables for the delay buffer rate:

- Enable RADIUS to modify the delay buffer rate based on service changes.

  a. Configure the attribute.

     ```
     [edit dynamic-profiles residential-silver variables]
     user@host# set dbrate radius vendor-id 4874 attribute 108
     ```

  b. Configure the tag.

     ```
     [edit dynamic-profiles residential-silver variables]
     user@host# set dbrate radius vendor-id 4874 tag 4
     ```

- Configure a default value for the delay buffer rate.

  ```
  [edit dynamic-profiles residential-silver variables]
  user@host# set dbrate default-value 10m
  ```

5. Do one of the following to configure variables for the scheduler map:

- Enable RADIUS to modify the scheduler map based on service changes.

  a. Configure the attribute.

     ```
     [edit dynamic-profiles residential-silver variables]
     user@host# set smap radius vendor-id 4874 attribute 108
     ```

  b. Configure the tag.

     ```
     [edit dynamic-profiles residential-silver variables]
     user@host# set smap radius vendor-id 4874 tag 1
     ```

- Configure a default value for the scheduler map.

  ```
  [edit dynamic-profiles residential-silver variables]
  user@host# set smap default-value triple-play
  ```

6. Configure the variables for the CoS parameters in the traffic-control profile.

   Either the shaping rate or the guaranteed rate is required in the traffic-control profile.

   a. Specify that you want to configure CoS parameters in the dynamic profile.

      ```
      user@host# edit dynamic-profiles residential-silver class-of-service
        traffic-control-profiles tcp1
      ```

   b. Configure the scheduler map variable.

      ```
      [edit dynamic-profiles residential-silver class-of-service traffic-control-profiles
        tcp1]
      user@host# set scheduler-map "$smap"
      ```

   c. Configure the shaping rate variable.

      ```
      [edit dynamic-profiles residential-silver class-of-service traffic-control-profiles
        tcp1]
      user@host# set shaping-rate "$srate"
      ```

   d. Configure the guaranteed rate variable.

      ```
      [edit dynamic-profiles residential-silver class-of-service traffic-control-profiles
        tcp1]
      user@host# set guaranteed-rate "$grate"
      ```

e. Configure the delay buffer rate variable.

```
[edit dynamic-profiles residential-silver class-of-service traffic-control-profiles
  tcp1]
user@host# set delay-buffer-rate "$dbrate"
```

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Changing CoS Services Overview on page 25

Applying CoS Traffic-Shaping Attributes to Dynamic Interface Sets and Member Subscriber Sessions

To control bandwidth at a household level in a subscriber access network, you can apply RADIUS dynamic class of service (CoS) traffic-shaping attributes to a dynamic interface set or agent-circuit-identifer (ACI) interface set and its member subscriber sessions when the member sessions are authenticated. The dynamic interface set or ACI interface set represents the *household* from which the subscriber sessions originate. The *subscriber sessions*, also referred to as *client sessions* or *subscriber interfaces*, can be dynamic VLAN, PPPoE, or IP demultiplexing (IP demux, for DHCP) subscriber interfaces.

To apply RADIUS dynamic CoS traffic-shaping attributes to both the dynamic interface set and its member subscriber sessions, you must configure two traffic-control profiles in the dynamic profile for the subscriber interface: one traffic-control profile for the “parent” dynamic interface set, and a second traffic-control profile for the dynamic subscriber interfaces. RADIUS tag values for the Junos OS CoS traffic shaping predefined variables used in both traffic-control profiles must be in the 100s range.

Before you begin:

- Create a dynamic profile that defines the VLAN, PPPoE, or IP demux logical subscriber interface.

  See the following topics:

  - *Configuring a Basic Dynamic Profile*
  - *Configuring VLAN Dynamic Profiles*
  - *Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles*
  - *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles*

To apply dynamic CoS traffic-shaping attributes to a dynamic ACI or non-ACI interface set and its member subscriber sessions in a dynamic profile for the subscriber interface:

1. Configure two traffic-control profiles at the **[edit dynamic-profiles *profile-name* class-of-service traffic-control profiles]** hierarchy level:

   - Traffic-control profile for the VLAN, PPPoE, or IP demux dynamic subscriber interfaces

- Traffic-control profile for the dynamic interface set or dynamic ACI interface set to which the subscriber interfaces belong

2. In the traffic-control profiles configured for the dynamic interface set and the subscriber interfaces, reference Junos OS CoS traffic-shaping predefined variables with RADIUS tag values in the 100s range.

   See "CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets" on page 73 for a complete list of the Junos OS predefined variables and RADIUS tag values that you must use in the traffic-control profiles for the dynamic subscriber interfaces and the dynamic interface set.

3. At the **[edit dynamic-profiles *profile-name* interfaces]** hierarchy level, use the **output-traffic-control-profile** statement to apply the traffic-control profiles to the dynamic subscriber interface and the dynamic interface set or dynamic ACI interface set.

Example: Dynamic PPPoE Subscriber Interface over Dynamic ACI Interface Set

The following example shows a dynamic profile named pppoe-subscriber that configures a dynamic PPPoE (**pp0**) subscriber interface over a dynamic ACI interface set.

The **traffic-control-profiles** stanza defines two traffic-control profiles: tcp-pppoe-session for the dynamic PPPoE subscriber interface, and tcp-parent-aci-set for the dynamic "parent" ACI interface set. The **$junos-cos-shaping-rate** predefined variable included in each of these traffic-control profiles is assigned RADIUS vendor ID 4874, attribute number 108, and tag value 102. The **$junos-cos-shaping-mode** variable is assigned RADIUS vendor ID 4874, attribute number 108, and tag value 107.

The **interfaces** stanza applies output traffic-control profile tcp-pppoe-session to the dynamic PPPoE (**pp0**) subscriber interface, and output traffic-control profile tcp-parent-aci-set to the dynamic ACI interface set.

```
[edit dynamic-profiles]
pppoe-subscriber {
  interfaces {
    interface-set "$junos-interface-set-name" {
      interface pp0 {
        unit "$junos-interface-unit";
      }
    }
    pp0 {
      unit "$junos-interface-unit" {
        ppp-options {
          pap;
        }
        pppoe-options {
          underlying-interface "$junos-underlying-interface";
          server;
        }
        no-keepalives;
```



```
            family inet {
              unnumbered-address lo0.0;
            }
          }
        }
      }
      class-of-service {
        traffic-control-profiles {
          tcp-pppoe-session {
            scheduler-map smap-1;
            shaping-rate $junos-cos-shaping-rate;
            overhead-accounting $junos-cos-shaping-mode frame-mode-bytes -4
              cell-mode-bytes 12;
          }
          tcp-parent-aci-set {
            shaping-rate $junos-cos-shaping-rate;
            overhead-accounting $junos-cos-shaping-mode frame-mode-bytes -4
              cell-mode-bytes 12;
          }
        }
        interfaces {
          pp0 {
            unit "$junos-interface-unit" {
              output-traffic-control-profile tcp-pppoe-session;
            }
          }
          interface-set $junos-interface-set-name {
            output-traffic-control-profile tcp-parent-aci-set;
          }
        }
      }
      }
    }
```

Related Documentation

- CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets on page 73
- CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions Overview on page 28
- Guidelines for Configuring CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions on page 30

CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets

To control bandwidth at a household level in a subscriber access network, you can apply RADIUS CoS traffic-shaping attributes to a dynamic interface set and its member subscriber sessions when the member sessions are authenticated. The dynamic interface set, which represents the household level in a subscriber access network, can be either a dynamic agent-circuit-identifier (ACI) interface set or a non-ACI–based dynamic interface set. The subscriber sessions belonging to the interface set can be dynamic VLAN, DHCP, or PPPoE subscriber interfaces.

To apply RADIUS CoS traffic-shaping attributes to both the dynamic interface set and its member subscriber sessions, you must configure two traffic-control profiles in the

dynamic profile for the subscriber interface: one traffic-control profile for the “parent” dynamic interface set, and a second traffic-control profile for the dynamic subscriber interfaces. RADIUS tag values for the Junos OS CoS traffic-shaping predefined variables used in these traffic-control-profiles must be in the 100s range, as described in Table 18 on page 74.

To accommodate this feature, the set of existing **$junos-cos-*parameter*** predefined dynamic variables for traffic shaping have been duplicated and assigned a tag value in the 100s range, as listed in Table 18 on page 74. The tag value is the only difference between the existing predefined dynamic variables and the predefined dynamic variables that you must use with this feature.

For example, the existing **$junos-cos-shaping-rate** predefined variable is assigned RADIUS vendor ID 4874, attribute number 108, and tag value 2. To apply RADIUS CoS traffic-shaping attributes to the dynamic interface set and its member subscriber sessions, you must instead use the **$junos-cos-shaping-rate** predefined variable that is assigned RADIUS vendor ID 4874, attribute number 108, and tag value 102.

Table 18 on page 74 describes the Junos OS predefined dynamic variables and RADIUS tag values that you can use in a dynamic profile to apply RADIUS CoS traffic-shaping attributes to the dynamic interface set and its member subscriber sessions. The table lists the predefined dynamic variables in ascending order by tag value.

NOTE: All of the predefined variables listed in Table 18 on page 74 use RADIUS vendor ID 4874 and RADIUS attribute value 108.

Table 18: Junos OS CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets

Predefined Variable	RADIUS Tag Value	Description
$junos-cos-scheduler-map	101	Scheduler-map name configured in a traffic-control profile in a dynamic profile.
$junos-cos-shaping-rate	102	Shaping rate configured in a traffic-control profile in a dynamic profile. Represents the maximum bandwidth of a CoS scheduler node.
$junos-cos-guaranteed-rate	103	Guaranteed rate configured in a traffic-control profile in a dynamic profile. Represents the minimum bandwidth of a CoS scheduler node.
$junos-cos-delay-buffer-rate	104	Delay-buffer rate configured in a traffic-control profile in a dynamic profile.

Table 18: Junos OS CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets *(continued)*

Predefined Variable	RADIUS Tag Value	Description
$junos-cos-excess-rate	105	Excess rate configured in a traffic-control profile in a dynamic profile; scheduler weighting when operating in the excess region between the guranteed rate and the shaping rate. NOTE: Do not configure the $junos-cos-excess-rate variable when either the $junos-cos-excess-rate-high variable or the $junos-cos-excess-rate-low variable is configured.
$junos-cos-traffic-control-profile	106	Traffic-control profile configured in a dynamic profile for subscriber access.
$junos-cos-shaping-mode	107	Overhead-accounting mode configured in a traffic-control profile in a dynamic profile to shape downstream ATM traffic based on either frames (frame-mode) or cells (cell-mode).
$junos-cos-byte-adjust	108	Byte adjustment value for the cell or frame shaping mode configured in a traffic-control profile in a dynamic profile. NOTE: Do not configure the $junos-cos-byte-adjust variable when either the $junos-cos-byte-adjust-frame variable or the $junos-cos-byte-adjust-cell variable is configured.
$junos-cos-adjust-minimum	109	Minimum adjusted shaping rate configured in a traffic-control profile for a dynamic subscriber interface. Specifying this variable in a traffic-control profile for a dynamic interface set has no effect.

Table 18: Junos OS CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets *(continued)*

Predefined Variable	RADIUS Tag Value	Description
$junos-cos-excess-rate-high	110	Shaping rate configured for excess high-priority traffic in a traffic-control profile in a dynamic profile. NOTE: Do not configure the $junos-cos-excess-rate-high variable when the $junos-cos-excess-rate variable is configured.
$junos-cos-excess-rate-low	111	Shaping rate configured for excess low-priority traffic in a traffic-control profile in a dynamic profile. NOTE: Do not configure the $junos-cos-excess-rate-low variable when the $junos-cos-excess-rate variable is configured.
$junos-cos-shaping-rate-burst	112	Burst size for the shaping rate configured in a traffic-control profile in a dynamic profile.
$junos-cos-guaranteed-rate-burst	113	Burst size for the guaranteed rate configured in a traffic-control profile in a dynamic profile.
$junos-cos-byte-adjust-frame	114	Overhead bytes when downstream ATM traffic is in frame-mode. NOTE: Do not configure the $junos-cos-byte-adjust-frame variable when the $junos-cos-byte-adjust variable is configured.
$junos-cos-byte-adjust-cell	115	Overhead bytes when downstream ATM traffic is in cell-mode. NOTE: Do not configure the $junos-cos-byte-adjust-cell variable when the $junos-cos-byte-adjust variable is configured.

Table 18: Junos OS CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets *(continued)*

Predefined Variable	RADIUS Tag Value	Description
$junos-cos-shaping-rate-priority-high	116	Shaping rate configured for high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-priority-high-burst	117	Shaping rate burst size configured for high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-priority-medium	118	Shaping rate configured for medium-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-priority-medium-burst	119	Shaping rate burst size configured for medium-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-priority-low	120	Shaping rate configured for low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.

Table 18: Junos OS CoS Traffic Shaping Predefined Variables for Dynamic Interface Sets *(continued)*

Predefined Variable	RADIUS Tag Value	Description
$junos-cos-shaping-rate-priority-low-burst	121	Shaping rate burst size configured for low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-excess-high	122	Shaping rate configured for excess high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-excess-high-burst	123	Shaping rate burst size configured for excess high-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-excess-low	124	Shaping rate configured for excess low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.
$junos-cos-shaping-rate-excess-low-burst	125	Shaping rate burst size configured for excess low-priority traffic in a traffic-control profile for a dynamic interface set or dynamic ACI interface set at a household level. Specifying this variable in a traffic-control profile for a dynamic subscriber interface is prohibited.

Related Documentation

- Applying CoS Traffic-Shaping Attributes to Dynamic Interface Sets and Member Subscriber Sessions on page 71
- CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions Overview on page 28

- Guidelines for Configuring CoS Traffic Shaping Attributes for Dynamic Interface Sets and Member Subscriber Sessions on page 30
- *Junos OS Predefined Variables*

CHAPTER 9

Configuration Tasks for Applying CoS to Interfaces

Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile

After you configure the traffic shaping and scheduling CoS parameters in a dynamic profile, you apply them to an interface. The output traffic-control profile enables you to provide traffic scheduling to the interface.

To apply CoS attributes to an interface in a dynamic profile:

1. Specify that you want to apply CoS attributes to an interface in the dynamic profile.

   user@host# **edit dynamic-profiles** ***profile-name*** class-of-service

2. Configure the interface name and logical interface using a variable, and apply the output traffic-control profile to the interface.

   [edit dynamic-profiles *profile-name* class-of-service interfaces]
   user@host# **set** interfaces **$junos-interface-ifd-name** unit
   $junos-underlying-interface-unit output-traffic-control-profile ***profile-name***

You can use one of the following methods to specify the output traffic-control profile you want to use:

- Reference the **$junos-cos-traffic-control-profile** predefined variable. At subscriber login, subscriber management takes one of the following actions, in the order listed:

  a. If RADIUS is being used and it returns a value for the traffic-control profile, subscriber management uses the RADIUS value.

  b. If RADIUS is not being used, subscriber management uses the default traffic-control profile (which is specified by the **predefined-variables-default** statement at the **[edit dynamic-profiles]** hierarchy).

     For example:

     ```
     user@host# set interfaces $junos-interface-ifd-name unit
       $junos-underlying-interface-unit output-traffic-control-profile
       $junos-cos-traffic-control-profile
     ```

- Explicitly reference the name of the traffic-control profile.

  For example:

  ```
  user@host# set interfaces $junos-interface-ifd-name unit
    $junos-underlying-interface-unit output-traffic-control-profile tcp-sales-2
  ```

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51
- Example: Configuring Static Hierarchical Scheduling and Queuing for Subscriber Access on page 101
- Example: Configuring Dynamic Hierarchical Scheduling and Queuing for Subscriber Access on page 103
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195
- CoS for Subscriber Access Overview on page 3

Applying Minimal Shaping and Scheduling to Remaining Subscriber Traffic

It is beneficial to apply a remaining traffic-control profile to a logical interface to provide minimal CoS scheduling when you have not configured or over-provisioned Layer 3 schedulers. In the event that schedulers are not available, the remaining subscriber traffic receives the essential level of service.

To configure scheduling for remaining subscriber traffic:

1. Enable hierarchical scheduling for the interface.

[edit interfaces *interface-name*]
user@host# **set** hierarchical-scheduler

2. Apply the remaining traffic-control profile to the port on which you enabled hierarchical scheduling.

   [edit class-of-service interfaces *interface-name*]
   user@host# **set output-traffic-control-profile-remaining** *profile-name*

Related Documentation

- Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile on page 81

Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile

Rewrite rules define the marking for various CoS values, including DSCP, DSCP IPv6, IP precedence, and IEEE 802.1 CoS values. Rewrite rules have an associated forwarding class and code-point alias or bit set.

NOTE: By default, subscriber lawful intercept does not intercept DHCP control packets that are generated by the routing engine. To ensure that a DHCP control packet generated by the routing engine is intercepted, you need to configure the ieee-802.1 rewrite-rule for VLAN demux.

For dynamic CoS, you define the rewrite rules mapping for the CoS values statically, then reference the rewrite rule configuration in the dynamic profile for the subscriber interface.

To configure a rewrite rule in a dynamic profile:

1. Define the rewrite-rules mapping for the traffic that passes through all queues on the interface. The available rewrite-rules types for dynamic CoS are **dscp**, **dscpv6**, **ieee-802.1** and **inet-precedence**.

   See *Configuring Rewrite Rules*.

2. Apply the rewrite-rules definition to the subscriber interface in the dynamic profile.

   [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number*]
   user@host# **edit** rewrite-rules

3. Configure the applicable rewrite rule markers in the dynamic profile.

   - For DSCP:

     [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]
     user@host# **set** dscp (*rewrite-name* | default)

   - For DSCPv6:

     [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]
     user@host# **set** dscp-ipv6 (*rewrite-name* | default)

- For IEEE 802.1:

  [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]
  user@host# **set** ieee-802.1 (*rewrite-name* | default) vlan-tag (outer | outer-and-inner)

- For inet-precedence:

  [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules]
  user@host# **set** inet-precedence (*rewrite-name* | default)

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Example: Configuring Dynamic Hierarchical Scheduling and Queuing for Subscriber Access on page 103
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *Applying IEEE 802.1p Rewrite Rules to Dual VLAN Tags*
- *Rewriting Packet Header Information Overview*

Applying a Classifier to a Subscriber Interface in a Dynamic Profile

You can apply the classification map to a subscriber interface in a dynamic profile.

For dynamic CoS, you define the classification map for the CoS values statically, then reference the classifier configuration in the dynamic profile for the subscriber interface.

To apply a classifier to an interface in a dynamic profile:

1. Define the classifier.

   The available classifier types for dynamic CoS are **dscp**, **dscp-ipv6**, **ieee-802.1**, and **inet-precedence**.

   See *Defining Classifiers*.

2. Apply the classifier definition to the subscriber interface in the dynamic profile.

   [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number*]
   user@host# **edit** classifiers

3. Configure the applicable classifiers in the dynamic profile.

   - For DSCP:

     [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers]
     user@host# **set** dscp (*classifier-name* | default)

   - For DSCPv6:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name unit
  logical-unit-number classifiers]
user@host# set dscp-ipv6 (classifier-name | default)
```

- For IEEE 802.1:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name unit
  logical-unit-number classifiers]
user@host# set ieee-802.1 (classifier-name | default) vlan-tag (inner | outer)
```

- For inet-precedence:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name unit
  logical-unit-number classifiers]
user@host# set inet-precedence (classifier-name | default)
```

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Example: Configuring Dynamic Hierarchical Scheduling and Queuing for Subscriber Access on page 103
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195
- Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile on page 83
- *Overview of BA Classifier Types*
- *Default Behavior Aggregate Classification Overview*

Configuring an Interface Set of Subscribers in a Dynamic Profile

Interface sets enable you to provide hierarchical scheduling to a group of subscriber interfaces.

Before you begin, configure the subscriber interfaces that you intend to include in the interface set.

- For static VLAN interfaces, see *Configuring Static Subscriber Interfaces in Dynamic Profiles*.
- For dynamic VLAN interfaces, see *Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet*.
- For dynamic IP demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles* and *Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet*.
- For dynamic VLAN demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles*.
- For dynamic PPPoE interfaces, see *Configuring Dynamic PPPoE Subscriber Interfaces Using Dynamic Profiles*.
- For aggregated Ethernet interfaces, see "Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links" on page 86

To configure an interface set of subscriber interfaces:

1. Configure the interface set in the dynamic profile.

   [edit dynamic-profiles *profile-name* interfaces]
   user@host#edit interface-set ***interface-set-name***

   Replacing the ***interface-set-name*** variable with the **$junos-interface-set-name**, **$junos-svlan-interface-set-name**, or **$junos-tagged-vlan-interface-set-name** predefined variable. The interface set is created dynamically when the subscriber logs in.

2. Include the interfaces within the dynamic interface-set.

   [edit dynamic-profiles *profile-name* interfaces interface-set $junos-interface-set-name]
   user@host# set interface ***interface-name*** unit ***logical-unit-number***

3. Apply traffic shaping and queuing parameters to the interface set.

   TIP: You must configure the interface set in the static **[edit class-of-service]** hierarchy, not in the **[edit dynamic-profiles]** hierarchy.

   [edit class-of-service interfaces]
   user@host# edit interface-set *interface-set-name*
   [edit class-of-service interfaces interface-set *interface-set-name*]
   user@host# set output-traffic-control-profile *profile-name*

Related Documentation

- CoS for Interface Sets of Subscribers Overview on page 17
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for Interface Sets of Subscribers Overview on page 17
- Example: Configuring a Dynamic Interface Set of VLAN Subscribers on page 130
- CoS for Aggregated Ethernet Subscriber Interfaces Overview on page 11

Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links

You can enable hierarchical CoS on a subscriber interface with an underlying aggregated Ethernet interface.

Before you begin, configure the subscriber interface with aggregated Ethernet.

- To configure a VLAN interface over aggregated Ethernet with link protection, see *Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet* and *Configuring Link Protection for Aggregated Ethernet Interfaces*.
- To configure a demux subscriber interface:

  For static and dynamic IP demux interfaces, see *Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet*.

  For static and dynamic VLAN demux interfaces, see *Configuring a Static or Dynamic VLAN Demux Subscriber Interface over Aggregated Ethernet*.

BEST PRACTICE: Link protection is not required for IP or demux subscriber interfaces. We recommend that you enable targeted distribution on the demux interface to provide accurate hierarchical scheduling for these links. See *Providing Accurate Scheduling for a Demux Subscriber Interface of Aggregated Ethernet Links*.

To configure hierarchical CoS on the link aggregation (LAG) bundle:

1. Specify that you want to access the LAG bundle.

   user@host# edit interfaces ae*x*

2. Configure the link aggregation (LAG) bundle with hierarchical scheduler mode.

   [edit interfaces ae*x*]
   user@host# set hierarchical-scheduler

You can then attach static or dynamic traffic shaping and scheduling parameters at the aggregated Ethernet logical interface or its underlying physical interface. See:

- Configuring Traffic Scheduling and Shaping for Subscriber Access on page 57
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile on page 81

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195
- CoS for Subscriber Access Overview on page 3
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4

Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface

You can configure hierarchical CoS on a static PPPoE subscriber interface.

Before you begin:

- Configure the static PPPoE subscriber interface.

  See *Configuring PPPoE*.

To configure hierarchical CoS on a static PPPoE subscriber interface:

1. Specify the PPPoE interface that you want to configure.

   user@host# edit interfaces *pppoe-interface-name*

2. Configure the hierarchical scheduler for the interface.

   [edit interfaces *interface-name*]

```
user@host# set hierarchical-scheduler
```

3. (Optional) Group the PPPoE interfaces in an interface set.

```
[edit]
user@host# edit interfaces interface-set interface-set-name
```

You can now configure static traffic and scheduling parameters for each traffic-control profile, and attach each traffic-control profile to the PPPoE interface or the PPPoE interface set. For more information, see *Using the CLI to Modify Traffic-Control Profiles That Are Currently Applied to Subscribers*.

Related Documentation

- For hardware requirements and configuration guidelines, see Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for PPPoE Subscriber Interfaces Overview on page 12
- Example: Configuring Hierarchical Scheduling and Queuing for a Static PPPoE Subscriber Interface on page 123
- Example: Configuring Hierarchical Scheduling and Queuing for an Underlying Static PPPoE Subscriber Interface on page 126
- Example: Configuring Hierarchical Scheduling and Queuing for an Interface Set of Static PPPoE Subscriber Interfaces on page 128
- Verifying the Scheduling and Shaping Configuration for Subscriber Access on page 195
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4

Configuring Dynamic CoS for an L2TP LAC Tunnel

In L2TP configurations, IP and L2TP headers are added to packets arriving at a PPP subscriber interface on the LAC before being tunneled to the L2TP network server (LNS).

Classifiers and rewrite rules enable you to properly transfer the ToS (Type of Service) value or the 802.1p value from the inner IP header to the outer IP header of the L2TP packet.

Before you begin, configure the L2TP LAC. See *Configuring an L2TP LAC*.

To manage the IP header values for a LAC tunnel:

1. Configure the classifier for the inner tunnel.
   a. Define the fixed or behavior aggregate (BA) classifier.
      - To configure a fixed classifier:

        ```
        [edit class-of-service interfaces interface-name unit logical-unit-number]
        user@host# set forwarding-class class-name
        ```

      - To configure a BA classifier:

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

```
user@host#set classifiers (ieee-802.1 | inet-precedence) classifier-name
  forwarding-class class-name loss-priority level code-points [ aliases ] [
  bit-patterns]
```

b. Apply the classifier to the Layer 2 interface or Layer 3 interface. For Layer 2, you can apply the classifier at the PPP interface or an underlying VLAN interface. For Layer 3, you can apply classifiers to a family of PPP interfaces.

- To apply the classifier for the IEEE 802.1p value:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name
  unit logical-unit-number classifiers]
user@host# set ieee-802.1 (classifier-name | default) vlan-tag (inner | outer)
```

- To apply the classifier for the ToS value:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name
  unit logical-unit-number classifiers]
user@host# set inet-precedence (classifier-name | default)
```

2. Configure the rewrite rule for the egress tunnel.

a. Configure the rewrite rule with the forwarding class and the loss priority value.

```
[edit class-of-service]
user@host# set rewrite-rules (ieee-802.1 | inet-precedence) rewrite-name
  forwarding-class class-name loss-priority level code-point (alias | bits)
```

b. Apply the rewrite rule to the PPP interface for which the L2TP tunnel is configured.

- To apply the rewrite-rule for the IEEE 802.1p value:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name
  unit logical-unit-number rewrite-rules]
user@host# set ieee-802.1 (rewrite-name | default) vlan-tag (outer |
  outer-and-inner)
```

- To apply the rewrite rule for the ToS value:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name
  unit logical-unit-number rewrite-rules]
user@host# set inet-precedence (rewrite-name | default)
```

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for L2TP LAC Subscriber Interfaces Overview on page 13

Configuring Dynamic CoS for an L2TP LNS Inline Service

You can configure hierarchical scheduling for an L2TP LNS inline service and manage the IP header values using rewrite rules and classifiers.

Before you begin, configure the L2TP LNS inline service interface. See *Configuring an L2TP LNS with Inline Service Interfaces*.

To configure CoS for an L2TP LNS inline service in a dynamic profile:

1. Configure the hierarchical scheduler for the service interface (**si**) interface.

   [edit interfaces si-*fpc/port/pic*]
   user@host# set hierarchical-scheduler maximum-hierarchy-levels 2

   BEST PRACTICE: To enable Level 3 nodes in the LNS scheduler hierarchy and to provide better scaling, we recommend that you also specify a maximum of two hierarchy levels.

2. Configure the LNS to reflect the IP ToS value in the inner IP header to the outer IP header.

   [edit services l2tp tunnel-group *name*]
   user@host# set tos-reflect

3. Configure the classifier for egress traffic from the LAC.

   a. Define the fixed or behavior aggregate (BA) classifier.

      - To configure a fixed classifier:

        [edit class-of-service interfaces *interface-name* unit *logical-unit-number*]
        user@host# set forwarding-class *class-name*

      - To configure a BA classifier:

        [edit class-of-service]
        user@host# set classifiers (dscp | dscp-ipv6 | inet-precedence) *classifier-name* forwarding-class *class-name* loss-priority *level* code-points [*aliases*] [*bit-patterns*]

   b. Apply the classifier to the service interface.

      - To apply the classifier for the DSCP or DSCP IPv6 value:

        [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers]
        user@host# set dscp (*classifier-name* | default)
        user@host# set dscp-ipv6 (*classifier-name* | default)

      - To apply the classifier for the ToS value:

        [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers]
        user@host# set inet-precedence (*classifier-name* | default)

4. Configure and apply a rewrite-rule to ingress traffic to the LAC:

   a. Configure the rewrite rule with the forwarding class and the loss priority value.

      [edit class-of-service]
      user@host# set rewrite-rules (dscp | dscp-ipv6 | inet-precedence) *rewrite-name* forwarding-class *class-name* loss-priority *level* code-point (*alias* | *bits*)

   b. Apply the rewrite rule to the service interface.

      - To apply the rewrite rule for the DSCP or DSCP IPv6 value:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name
  unit logical-unit-number rewrite-rules]
user@host# setdscp (rewrite-name | default)
user@host# set dscp-ipv6 (rewrite-name | default)
```

- To apply the rewrite rule for the ToS value:

```
[edit dynamic-profiles profile-name class-of-service interfaces interface-name
  unit logical-unit-number rewrite-rules]
user@host# set inet-precedence (rewrite-name | default)
```

5. (Optional) Configure additional adjustments for downstream ATM traffic.

   By default, the shaping calculation on the service interface includes the L2TP encapsulation. If necessary, you can configure additional adjustments for downstream ATM traffic from the LAC or differences in Layer 2 protocols.

```
[edit dynamic-profiles profile-name class-of-service traffic-control-profiles profile-name]
user@host# set overhead-accounting (frame-mode | cell-mode |
  $junos-cos-shaping-mode) <bytes (byte-value | $junos-cos-byte-adjust)
```

6. Apply the traffic-control profile.

```
[edit dynamic-profiles profile-name class-of-service interfaces
  $junos-interface-ifd-name unit $junos-interface-unit]
user@host# set output-traffic-control-profile profile-name
```

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- CoS for L2TP LNS Inline Services Overview on page 15
- *Example: Configuring an L2TP LNS*
- *Configuring Dynamic Shaping Parameters to Account for Overhead in Downstream Traffic Rates*

Applying CoS Attributes to VLANs Using Agent-Circuit-Identifiers

To apply CoS attributes, such as shaping, at the household level, you must set and define the CoS policy for the agent-circuit-identifier VLAN interface set using the dynamic profile for the agent-circuit-identifier interface set (not the subscriber profile). You can also configure a traffic-control profile and a remaining traffic-control profile for a dynamic interface set.

The following example is a CoS profile for an ACI set using a unique-ID based dynamic scheduler map:

Before you apply CoS attributes to VLANs:

- Create a basic dynamic profile.

  See *Configuring a Basic Dynamic Profile*.

Configure a CoS dynamic profile with a simple traffic-control profile that is applied to the dynamic interface set that represents the ACI VLAN.

1. Configure CoS to support a dynamic interface set in the CoS profile:

   ```
   [edit dynamic-profiles profile-name]
   user@host# edit interface "$junos-interface-name"
   ```

2. Configure the interfaces.

   ```
   [edit dynamic-profiles profile-name interfaces]
   user@host# edit interface-set "$junos-interface-set-name"
   user@host# edit interface "$junos-interface-ifd-name"
   ```

3. Configure the CoS traffic-control profile.

   ```
   [edit class-of-service]
   user@host# edit traffic-control-profiles traffic-control-profile-name
   user@host# set shaping-rate rate
   user@host# set guaranteed-rate rate
   ```

4. Specify the interfaces.

   ```
   [edit class-of-service interfaces]
   user@host# edit interface-set "$junos-interface-set-name"
   user@host# edit output-traffic-control-profile profile-name
   ```

The following example is a CoS profile for an ACI set using a unique ID-based dynamic scheduler map:

```
aci-set-profile {
  variables {
    ds1q0q2DP uid;
    ds1q1q2DP uid;
    be1_dp uid;
    ef1_dp uid;
    af1_dp uid;
    nc1_dp uid;
  }
  interfaces {
    interface-set "$junos-interface-set-name" {
      interface "$junos-interface-ifd-name";
    }
  }
  class-of-service {
    traffic-control-profiles {
      tcp2 {
        inactive: scheduler-map ss1q0q1DP;
        shaping-rate 50m;
        guaranteed-rate 30m;
        overhead-accounting bytes -20;
      }
```

```
    tcp3 {
      scheduler-map "$ds1q1q2DP";
      shaping-rate 30m;
      guaranteed-rate 10m;
      overhead-accounting bytes -20;
    }
  }
  interfaces {
    interface-set "$junos-interface-set-name" {
      output-traffic-control-profile tcp2;
      output-traffic-control-profile-remaining tcp3;
    }
  }
  scheduler-maps {
    "$ds1q0q2DP" {
      forwarding-class be scheduler "$be1_dp";
      forwarding-class af scheduler "$af1_dp";
      forwarding-class nc scheduler "$nc1_dp";
    }
    "$ds1q1q2DP" {
      forwarding-class ef scheduler "$ef1_dp";
      forwarding-class af scheduler "$af1_dp";
      forwarding-class nc scheduler "$nc1_dp";
    }
  }
  schedulers {
    "$be1_dp" {
      transmit-rate percent 25;
      priority low;
      drop-profile-map loss-priority low protocol any drop-profile d3;
      drop-profile-map loss-priority medium-low protocol any drop-profile d2;
      drop-profile-map loss-priority medium-high protocol any drop-profile d1;
      drop-profile-map loss-priority high protocol any drop-profile d0;
    }
    "$ef1_dp" {
      transmit-rate percent 25;
      priority low;
      drop-profile-map loss-priority low protocol any drop-profile d3;
      drop-profile-map loss-priority medium-low protocol any drop-profile d2;
      drop-profile-map loss-priority medium-high protocol any drop-profile d1;
      drop-profile-map loss-priority high protocol any drop-profile d0;
    }
    "$af1_dp" {
      transmit-rate percent 25;
      priority low;
      drop-profile-map loss-priority low protocol any drop-profile d3;
      drop-profile-map loss-priority medium-low protocol any drop-profile d2;
      drop-profile-map loss-priority medium-high protocol any drop-profile d1;
      drop-profile-map loss-priority high protocol any drop-profile d0;
    }
    "$nc1_dp" {
      transmit-rate percent 25;
      priority low;
      drop-profile-map loss-priority low protocol any drop-profile d3;
      drop-profile-map loss-priority medium-low protocol any drop-profile d2;
      drop-profile-map loss-priority medium-high protocol any drop-profile d1;
```

```
            drop-profile-map loss-priority high protocol any drop-profile d0;
        }
    }
  }
}
```

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Changing CoS Services Overview on page 25

CHAPTER 10

Configuration Tasks for Applying Hierarchical CoS to Ethernet MPLS Subscriber Interfaces

Configuring CoS Two-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces

Before configuring CoS parameters for MPLS pseudowire subscriber interfaces, you must first complete these tasks:

1. Configure the pseudowire logical interfaces. See *Configuring a Pseudowire Subscriber Logical Interface*.
2. Configure the pseudowire device count. See *Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router*.
3. Configure the pseudowire device including the logical tunnel anchor point. See *Configuring a Pseudowire Subscriber Logical Interface Device*.
4. Configure the pseudowire transport logical interface. See *Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface*.
5. Configure the pseudowire signaling (either Layer 2 circuit signaling or Layer 2 VPN signaling). See *Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces* or *Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces*.
6. Configure the pseudowire logical interfaces. See *Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface*.

To configure CoS policies on MPLS pseudowire subscriber interfaces using two-level scheduling:

1. Configure the hierarchical scheduler for the physical interface used for the logical tunnel (anchor point). For two-level scheduling the hierarchical scheduler must be set to **maximum-scheduler levels 2**.

   ```
   [edit]
   user@host#edit interfaces ps ps-anchor-device-name
   user@host#set hierarchical-scheduler maximum-hierarchy-levels 2
   ```

2. Specify the traffic-control profile to use on the pseudowire logical interface.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#set output-traffic-control-profile profile-name
   ```

3. Configure the rewrite rule.

   The available rewrite rule types for pseudowire interfaces are **dscp** and **inet-precedence**.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#edit rewrite-rules (dscp | inet-precedence) rewrite-name
   user@host#edit forwarding-class class-name
   user@host#set loss-priority class-name code-point (alias | bits)
   ```

4. Configure the classifier.

   The available classifier types for pseudowire interfaces are **dscp** and **inet-precedence**.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#edit classifiers (dscp | inet-precedence) classifier-name
   user@host#edit forwarding-class class-name
   user@host#set loss-priority class-name code-points [aliases] [bit-patterns]
   ```

5. Apply the rewrite rule and classifier to the pseudowire interface.

   For the *interface_name* parameter, specify the pseudowire device name.

   ```
   [edit class-of-service interfaces interface_name unit logical-unit-number]
   user@host#set rewrite-rule (dscp | inet-precedence) (rewrite-name | default) protocol
     protocol-types
   user@host#set classifiers (dscp | inet-precedence) (classifier-name | default)
   ```

Related Documentation

- *CoS on Ethernet Pseudowires in Universal Edge Networks Overview*
- *CoS Inputs and Outputs Examples*
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31
- CoS Two-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 32

- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 34
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 37
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface)

Before configuring CoS three-level scheduling on pseudowire logical interfaces over a transport logical interface, you must first complete these tasks:

1. Configure the pseudowire logical interfaces. See *Configuring a Pseudowire Subscriber Logical Interface*.
2. Configure the pseudowire device count. See *Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router*.
3. Configure the pseudowire device including the logical tunnel anchor point. See *Configuring a Pseudowire Subscriber Logical Interface Device*.
4. Configure the pseudowire transport logical interface. See *Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface*.
5. Configure the pseudowire signaling (either Layer 2 circuit signaling or Layer 2 VPN signaling). See *Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces* or *Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces*.
6. Configure the pseudowire logical interfaces. See *Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface*.

Three-level scheduling on pseudowire logical interfaces over a transport logical interface requires you to apply the traffic-control profiles at both the pseudowire logical interface and the pseudowire transport logical interface. To configure CoS policies on three-level scheduling on pseudowire logical interfaces over a transport logical interface:

1. Configure the hierarchical scheduler for the physical interface used for the logical tunnel (anchor point). For three-level scheduling the hierarchical scheduler must be set to **implicit-hierarchy**.

   ```
   [edit]
   user@host#edit interfaces ps-anchor-device-name
   user@host#set hierarchical-scheduler implicit-hierarchy
   ```

2. Specify the traffic-control profile to use on the pseudowire logical interface.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#set output-traffic-control-profile profile-name
   ```

3. Specify the traffic-control profile to use on the pseudowire transport logical interface.

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



```
user@host#edit interfaces ps ps-device-name
user@host#edit unit logical-unit-number
user@host#set output-traffic-control-profile profile-name
```

4. Configure the rewrite rule.

   The available rewrite rule types for pseudowire interfaces are **dscp** and **inet-precedence**.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#edit rewrite-rules (dscp | inet-precedence) rewrite-name
   user@host#edit forwarding-class class-name
   user@host#set loss-priority class-name code-point (alias | bits)
   ```

5. Configure the classifier.

   The available classifier types for pseudowire interfaces are **dscp** and **inet-precedence**.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#edit classifiers (dscp | inet-precedence) classifier-name
   user@host#edit forwarding-class class-name
   user@host#set loss-priority class-name code-points [aliases] [bit-patterns]
   ```

6. Apply the rewrite rule and classifier to the pseudowire interfaces.

   For the *interface_name* parameter, specify the pseudowire device name.

   ```
   [edit class-of-service interfaces interface_name unit logical-unit-number]
   user@host#set rewrite-rule (dscp | inet-precedence) (rewrite-name | default) protocol
     protocol-types
   user@host#set classifiers (dscp | inet-precedence) (classifier-name | default)
   ```

Related Documentation

- *CoS on Ethernet Pseudowires in Universal Edge Networks Overview*
- *CoS Inputs and Outputs Examples*
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 34
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 37
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set) on page 99
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Pseudowire Interface Set)

Before configuring three-level scheduling on pseudowire logical interfaces over a pseudowire logical interface set, you must first complete the following tasks:

1. Configure the pseudowire logical interfaces. See *Configuring a Pseudowire Subscriber Logical Interface*.
2. Configure the pseudowire device count. See *Configuring the Maximum Number of Pseudowire Logical Interface Devices Supported on the Router*.
3. Configure the pseudowire device including the logical tunnel anchor point. See *Configuring a Pseudowire Subscriber Logical Interface Device*.
4. Configure the pseudowire transport logical interface. See *Configuring the Transport Logical Interface for a Pseudowire Subscriber Logical Interface*.
5. Configure the pseudowire signaling (either Layer 2 circuit signaling or Layer 2 VPN signaling). See *Configuring Layer 2 Circuit Signaling for Pseudowire Subscriber Logical Interfaces* or *Configuring Layer 2 VPN Signaling for Pseudowire Subscriber Logical Interfaces*.
6. Configure the pseudowire logical interfaces. See *Configuring the Service Logical Interface for a Pseudowire Subscriber Logical Interface*.

Three-level scheduling on pseudowire logical interfaces over a pseudowire logical interface set requires you to apply the traffic-control profiles at both the pseudowire logical interface and the pseudowire logical interface-set. To configure CoS policies on MPLS pseudowire subscriber interfaces using three-level implicit hierarchical scheduling:

1. Configure the hierarchical scheduler for the physical interface used for the logical tunnel (anchor point). For three-level scheduling the hierarchical scheduler must be set to **implicit-hierarchy**.

   ```
   [edit]
   user@host#edit interfaces ps-anchor-device-name
   user@host#set hierarchical-scheduler implicit-hierarchy
   ```

2. Specify the traffic-control profile to use on the pseudowire logical interfaces.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#set output-traffic-control-profile profile-name
   ```

3. Define a pseudowire logical interface set and configure the traffic-control profile used for the interface set.

   ```
   [edit class-of-service]
   user@host#edit interfaces
   user@host#edit interface-set interface-set-name
   user@host#edit output-traffic-control-profile profile-name
   ```

4. Group the pseudowire logical interfaces in the pseudowire logical interface set.

```
[edit ]
user@host#edit interfaces
user@host#edit interface-set interface-set-name
user@host#edit interface ps ps-device-name
user@host#edit unit logical-unit-number
```

5. Configure the rewrite rule.

   The available rewrite rule types for pseudowire interfaces are **dscp** and **inet-precedence**.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#edit rewrite-rules (dscp | inet-precedence) rewrite-name
   user@host#edit forwarding-class class-name
   user@host#set loss-priority class-name code-point (alias | bits)
   ```

6. Configure the classifier.

   The available classifier types for pseudowire interfaces are **dscp** and **inet-precedence**.

   ```
   [edit class-of-service]
   user@host#edit interfaces ps ps-device-name
   user@host#edit unit logical-unit-number
   user@host#edit classifiers (dscp | inet-precedence) classifier-name
   user@host#edit forwarding-class class-name
   user@host#set loss-priority class-name code-points [aliases] [bit-patterns]
   ```

7. Apply the rewrite rule and classifier to the pseudowire interfaces.

   For the *interface_name* parameter, specify the ps device name.

   ```
   [edit class-of-service interfaces interface_name unit logical-unit-number]
   user@host#set rewrite-rule (dscp | inet-precedence) (rewrite-name | default) protocol
     protocol-types
   user@host#set classifiers (dscp | inet-precedence) (classifier-name | default)
   ```

Related Documentation

- *CoS on Ethernet Pseudowires in Universal Edge Networks Overview*
- *CoS Inputs and Outputs Examples*
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31
- CoS Three-Level Hierarchical Scheduling on MPLS Pseudowire Subscriber Interfaces on page 34
- CoS Configuration Overview for MPLS Pseudowire Subscriber Interfaces on page 37
- Configuring CoS Three-Level Hierarchical Scheduling for MPLS Pseudowire Subscriber Interfaces (Logical Interfaces over a Transport Logical Interface) on page 97
- **hierarchical-scheduler (Subscriber Interfaces on MX Series Routers) on page 168**

CHAPTER 11

Examples

Example: Configuring Static Hierarchical Scheduling and Queuing for Subscriber Access

This example shows you how to configure CoS for a subscriber in a dynamic profile. The CoS parameters configure a best-effort, data service for subscribers.

1. Configure the static CoS parameters in the **[edit class-of-service]** hierarchy.

   You must configure the scheduler maps in this hierarchy; it will get referenced in the dynamic profile.

   ```
   class-of-service {
     forwarding-classes {
       queue 0 best-effort;
       queue 1 expedited-forwarding;
   ```

```
    queue 3 network-control;
    queue 2 assured-forwarding;
  }
  scheduler-maps {
    data_smap {
      forwarding-class best-effort scheduler be_sch;
    }
  }
  schedulers {
    be_sch {
      transmit-rate percent 10;
      buffer-size remainder;
      priority low;
    }
  }
}
```

2. Configure the subscriber interface in the **[edit interfaces]** hierarchy. Enable hierarchical scheduling for the interface.

```
interfaces {
  ge-2/2/0 {
    hierarchical-scheduler;
    vlan-tagging;
    unit 100 {
      vlan-id 100;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 100.0.0.1;
      }
    }
  }
}
```

3. Configure CoS in the dynamic profile.

```
dynamic-profiles {
  data-service {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit" {
          family inet;
        }
      }
    }
    class-of-service {
      traffic-control-profiles {
        tcp1 {
          scheduler-map data_smap;
          shaping-rate 50k;
          guaranteed-rate 10k;
        }
      }
      interfaces {
        "$junos-interface-ifd-name" {
          unit "$junos-underlying-interface-unit" {
            output-traffic-control-profile tcp1;
          }
```



```
        }
      }
    }
  }
}
```

Related Documentation

- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Changing CoS Services Overview on page 25
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41

Example: Configuring Dynamic Hierarchical Scheduling and Queuing for Subscriber Access

In this example, subscribers are provided with a data and voice service defined in an access profile when they initially log in. The RADIUS administrator supplies the initial values on the RADIUS server, and the service activation is performed at subscriber login.

After the initial login, the subscriber adds an assured forwarding service that is not defined in the original access profile. A service profile is used to configure the schedulers and a RADIUS CoA activates the service. The queues defined for the schedulers in the initial scheduler map and the new scheduler map are merged.

In addition, the values for the initial data and voice service are upgraded by the RADIUS administrator through a separate RADIUS CoA message.

To configure the initial service and enable the activation through a RADIUS CoA:

1. Configure the access profile for the service activation.

   a. Configure the VLAN interface for the access profile.

      ```
      [edit]
      dynamic-profiles access-profile {
        interfaces {
          $junos-interface-ifd-name {
            unit $junos-underlying-interface-unit {
              family inet;
            }
          }
        }
      }
      ```

   b. Configure the class of service parameters in the access profile. In this example, you configure Junos OS predefined variables that provide the initial scheduler name and scheduler parameters obtained from the RADIUS authentication server when the subscriber logs in.

      Include the configurations for the interfaces, schedulers, and the scheduler maps.

      ```
      [edit]
      dynamic-profiles access-profile {
        class-of-service {
      ```

```
traffic-control-profiles {
  tcp1 {
    scheduler-map $junos-cos-scheduler-map;
    shaping-rate $junos-cos-shaping-rate;
    guaranteed-rate $junos-cos-guaranteed-rate;
    delay-buffer-rate $junos-cos-delay-buffer-rate;
  }
}
interfaces {
  $junos-interface-ifd-name {
    unit "$junos-underlying-interface-unit" {
      classifiers {
        ieee-802.1 l2_classifier;
      }
      rewrite-rules {
        ieee-802.1 l2_rewrite;
      }
    output-traffic-control-profile tcp1;
    }
  }
}
schedulers {
  $junos-cos-scheduler {
    buffer-size percent $junos-cos-scheduler-bs;
    priority $junos-cos-scheduler-pri;
    transmit-rate percent $junos-cos-scheduler-tx;
    drop-profile-map loss-priority low protocol any $junos-cos-scheduler-low;
    drop-profile-map loss-priority medium-low protocol any
      $junos-cos-scheduler-medium-low;
    drop-profile-map loss-priority medium-high protocol any
      $junos-cos-scheduler-medium-high;
    drop-profile-map loss-priority high protocol any $junos-cos-scheduler-high;
  }
}
scheduler-maps {
  data_voice_smap {
    forwarding-class be scheduler be_sch;
    forwarding-class ef scheduler ef_sch;
  }
}
}
}
```

Table 19 on page 104 lists the initial values defined by the RADIUS administrator for the scheduler map and shaping rates.

Table 19: Initial Scheduler Map and Shaping Values at Subscriber Login

Predefined Variable	RADIUS Tag	Value
$junos-cos-scheduler-map	T01	data_voice_smap
$junos-cos-shaping-rate	T02	6m
$junos-cos-guaranteed-rate	T03	4m

Table 19: Initial Scheduler Map and Shaping Values at Subscriber Login *(continued)*

Predefined Variable	RADIUS Tag	Value
$junos-cos-delay-buffer-rate	T04	4m

Table 20 on page 105 lists the initial values defined by the RADIUS administrator for the voice (expedited forwarding) scheduler.

Table 20: Initial CoS Values for the Voice Scheduler at Subscriber Login

Predefined Variable	Tag	Value
$junos-cos-scheduler	—	ef_sch
$junos-cos-scheduler-tx	T01	10
$junos-cos-scheduler-bs	T02	10
$junos-cos-scheduler-pri	T03	medium-high
$junos-cos-scheduler-dropfile-low	T04	d3
$junos-cos-scheduler-dropfile-medium-low	T05	d2
$junos-cos-scheduler-dropfile-medium-high	T06	d1
$junos-cos-scheduler-dropfile-high	T07	d0

Table 21 on page 105 lists the initial values defined by the RADIUS administrator for the data (best effort) scheduler.

Table 21: Initial CoS Values for the Data Scheduler at Subscriber Login

Predefined Variable	Tag	Value
$junos-cos-scheduler	—	be_sch
$junos-cos-scheduler-tx	T01	10
$junos-cos-scheduler-bs	T02	10
$junos-cos-scheduler-pri	T03	low
$junos-cos-scheduler-dropfile-low	T04	d0
$junos-cos-scheduler-dropfile-medium-low	T05	d1
$junos-cos-scheduler-dropfile-medium-high	T06	d2
$junos-cos-scheduler-dropfile-high	T07	d3

2. Configure the classifiers, drop profiles, forwarding classes, and rewrite rules in the static **[edit class-of-service]** hierarchy.

```
[edit]
class-of-service {
  classifiers {
    dscp dscp_classifier {
      forwarding-class be {
        loss-priority low code-points 000000;
      }
      forwarding-class af {
        loss-priority medium-low code-points 000001;
      }
    }
    ieee-802.1 l2_classifier {
      forwarding-class be {
        loss-priority medium-low code-points 000;
      }
      forwarding-class ef {
        loss-priority medium-low code-points 100;
      }
      forwarding-class af {
        loss-priority medium-low code-points 010;
      }
    }
  }
  drop-profiles {
    d0 {
      fill-level 25 drop-probability 100;
      fill-level 0 drop-probability 0;
    }
    d1 {
      fill-level 50 drop-probability 100;
      fill-level 0 drop-probability 0;
    }
    d2 {
      fill-level 75 drop-probability 100;
      fill-level 0 drop-probability 0;
    }
    d3 {
      fill-level 0 drop-probability 0;
      fill-level 100 drop-probability 100;
    }
  }
  forwarding-classes {
    queue 0 be;
    queue 1 ef;
    queue 2 af;
    queue 3 nc;
  }
  interfaces {
    ge-1/2/9 {
      shaping-rate 100m;
    }
  }
  rewrite-rules {
```

```
    ieee-802.1 l2_rewrite {
      forwarding-class be {
        loss-priority medium-low code-point 000;
      }
      forwarding-class ef {
        loss-priority medium-low code-point 001;
      }
      forwarding-class af {
        loss-priority medium-low code-point 100;
      }
    dscp l2_rewrite {
      forwarding-class be {
        loss-priority medium-low code-points 000;
      }
      forwarding-class ef {
        loss-priority medium-low code-points 001;
      }
      forwarding-class af {
        loss-priority medium-low code-points 001;
      }
  }
}
```

3. Configure the service profile enable RADIUS to activate the video service after login. The video service corresponds to assured forwarding PHB.

   In this example, you configure Junos OS predefined variables that provide the initial scheduler name and scheduler parameters obtained from the RADIUS authentication server when the subscriber logs in.

```
[edit]
dynamic-profiles service-af {
  variables {
    af_fc default-value video;
    af_sch default-value af_sch;
    sch-drop-any default-value all;
    sch-pri-2 default-value strict-high;
    sch-bs-2 default-value 40;
    sch-tx-2 default-value 3m;
    smap default-value any
  }
    class-of-service {
      scheduler-maps {
        "$smap" {
          forwarding-class “$af_fc” scheduler “$af_sch”;
        }
      }
      schedulers {
        "$af_sch" {
          transmit-rate percent "$sch-tx-2";
          buffer-size percent "$sch-bs-2";
          priority "$sch-pri-2";
          drop-profile-map loss-priority any protocol any drop-profile “$sch-drop-any”;
        }
      }
    }
```

```
}
```

After the three services are activated, subscribers receive upgraded values for the data and voice service when RADIUS sends a change of authorization (CoA). In this case, the CoS parameters are replaced, because multiple subscribers were not enabled on the logical interface.

Table 22 on page 108 lists the upgraded values defined by the RADIUS administrator.

Table 22: Upgraded CoS Values for the Video Service

Variable	RADIUS Tag	Value
junos-cos-scheduler-map	T01	data_voice_smap
junos-cos-shaping-rate	T02	14m
junos-cos-guaranteed-rate	T03	13m
junos-cos-delay-buffer-rate	T04	12m

Table 23 on page 108 lists the values defined by the RADIUS administrator for the video (assured forwarding) scheduler.

Table 23: Upgraded CoS Values for the Video Scheduler

Predefined Variable	Tag	Value
$junos-cos-scheduler	—	af_sch
$junos-cos-scheduler-tx	T01	10
$junos-cos-scheduler-bs	T02	10
$junos-cos-scheduler-pri	T03	medium
$junos-cos-scheduler-dropfile-low	T04	d3
$junos-cos-scheduler-dropfile-medium-low	T05	d2
$junos-cos-scheduler-dropfile-medium-high	T06	d1
$junos-cos-scheduler-dropfile-high	T07	d0

Table 24 on page 109 lists the values defined by the RADIUS administrator for the expedited forwarding scheduler in the CoA message. The values are the same as the initial service.

Table 24: Initial CoS Values for the Expedited Forwarding Scheduler at Subscriber Login

Predefined Variable	Tag	Value
$junos-cos-scheduler	—	ef_sch
$junos-cos-scheduler-tx	T01	10
$junos-cos-scheduler-bs	T02	10
$junos-cos-scheduler-pri	T03	medium-high
$junos-cos-scheduler-dropfile-low	T04	d3
$junos-cos-scheduler-dropfile-medium-low	T05	d2
$junos-cos-scheduler-dropfile-medium-high	T06	d1
$junos-cos-scheduler-dropfile-high	T07	d0

Table 25 on page 109 lists the values defined by the RADIUS administrator for the best effort scheduler in the CoA message. The values are the same as the initial service.

Table 25: Initial CoS Values for the Best Effort Scheduler at Subscriber Login

Predefined Variable	Tag	Value
$junos-cos-scheduler	—	be_sch
$junos-cos-scheduler-tx	T01	10
$junos-cos-scheduler-bs	T02	10
$junos-cos-scheduler-pri	T03	low
$junos-cos-scheduler-dropfile-low	T04	d0
$junos-cos-scheduler-dropfile-medium-low	T05	d1
$junos-cos-scheduler-dropfile-medium-high	T06	d2
$junos-cos-scheduler-dropfile-high	T07	d3

Related Documentation

- Changing CoS Services Overview on page 25
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4

Example: Configuring Per-Unit Scheduling for Subscriber Access

In this example, a network administrator sets up a subscriber access configuration with per-unit scheduling.

1. The administrator configures the static VLAN interfaces and enables per-unit scheduling for the interfaces.

```
[edit]
  interfaces {
  ge-1/1/0 {
    per-unit-scheduler;
    vlan-tagging;
    unit 100 {
      vlan-id 100;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
    unit 200 {
      vlan-id 200;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
  }
  ge-1/1/1 {
    per-unit-scheduler;
    vlan-tagging;
    unit 100 {
      vlan-id 100;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
    unit 200 {
      vlan-id 200;
      family inet {
        unnumbered-address lo0.0 preferred-source-address 192.100.1.1;
      }
    }
  }
  ge-1/0/1 {
    unit 0 {
      family inet {
        address 3.1.1.1/24;
      }
    }
  }
  ge-1/1/2 {
    description "wfce14 eth1 soso ge-1/1/2";
    vlan-tagging;
    gigether-options {
      no-auto-negotiation;
```

```
    }
    unit 100 {
      vlan-id 100;
      family inet {
        address 121.0.0.1/24;
      }
    }
  }
}
```

2. The administrator configures static CoS parameters, including forwarding classes and classifiers, to be referenced in the dynamic profiles.

```
[edit]
  class-of-service {
  classifiers {
    inet-precedence 8q-inet {
      forwarding-class be {
        loss-priority low code-points 000;
      }
      forwarding-class ef {
        loss-priority low code-points 001;
      }
      forwarding-class af {
        loss-priority low code-points 010;
      }
      forwarding-class nc {
        loss-priority low code-points 011;
      }
      forwarding-class voice {
        loss-priority low code-points 100;
      }
      forwarding-class video {
        loss-priority low code-points 101;
      }
      forwarding-class game {
        loss-priority low code-points 110;
      }
      forwarding-class data {
        loss-priority low code-points 111;
      }
    }
    inet-precedence 4q-inet {
      forwarding-class be {
        loss-priority low code-points [ 000 001 ];
      }
      forwarding-class ef {
        loss-priority low code-points [ 010 011 ];
      }
      forwarding-class af {
        loss-priority low code-points [ 100 101 ];
      }
      forwarding-class nc {
        loss-priority low code-points [ 110 111 ];
      }
    }
```

```
    inet-precedence 8q-drop-inet {
      forwarding-class be {
        loss-priority low code-points 000;
      }
      forwarding-class ef {
        loss-priority medium-low code-points 001;
      }
      forwarding-class af {
        loss-priority medium-high code-points 010;
      }
      forwarding-class nc {
        loss-priority high code-points 011;
      }
      forwarding-class voice {
        loss-priority low code-points 100;
      }
      forwarding-class video {
        loss-priority medium-low code-points 101;
      }
      forwarding-class game {
        loss-priority medium-high code-points 110;
      }
      forwarding-class data {
        loss-priority high code-points 111;
      }
    }
    inet-precedence 4q-drop-inet {
      forwarding-class be {
        loss-priority low code-points [ 000 001 ];
      }
      forwarding-class ef {
        loss-priority medium-low code-points [ 010 011 ];
      }
      forwarding-class af {
        loss-priority medium-high code-points [ 100 101 ];
      }
      forwarding-class nc {
        loss-priority high code-points [ 110 111 ];
      }
    }
  }
  drop-profiles {
    d0 {
      fill-level 25 drop-probability 100;
      fill-level 0 drop-probability 0;
    }
    d1 {
      fill-level 50 drop-probability 100;
      fill-level 0 drop-probability 0;
    }
    d2 {
      fill-level 75 drop-probability 100;
      fill-level 0 drop-probability 0;
    }
    d3 {
      fill-level 100 drop-probability 100;
```

```
            fill-level 0 drop-probability 0;
        }
        all {
            fill-level 0 drop-probability 0;
            fill-level 100 drop-probability 100;
        }
    }
    forwarding-classes {
        queue 0 be;
        queue 1 ef;
        queue 2 af;
        queue 3 nc;
        queue 4 voice;
        queue 5 video;
        queue 6 game;
        queue 7 data;
    }
    interfaces {
        ge-1/0/1 {
            unit 0 {
                classifiers {
                    inet-precedence 8q-drop-low-high-inet;
                }
            }
        }
    }
    traceoptions {
        flag all;
        flag asynch;
        flag route-socket;
    }
}
```

3. The administrator configures the access and service dynamic profiles to receive CoS parameters for the subscriber interfaces through RADIUS.

```
[edit]
    dynamic-profiles {
    subscriber {
        interfaces {
            "$junos-interface-ifd-name" {
                unit "$junos-underlying-interface-unit" {
                    family inet;
                }
            }
        }
        class-of-service {
            traffic-control-profiles {
                zero {
                    scheduler-map "$junos-cos-scheduler-map";
                    shaping-rate "$junos-cos-shaping-rate";
                    guaranteed-rate "$junos-cos-guaranteed-rate";
                    delay-buffer-rate "$junos-cos-delay-buffer-rate";
                }
            }
            interfaces {
```

```
        "$junos-interface-ifd-name" {
          unit "$junos-underlying-interface-unit" {
            output-traffic-control-profile zero;
          }
        }
      }
      scheduler-maps {
        be_smap {
          forwarding-class be scheduler be_sch;
        }
        all_smap {
          forwarding-class be scheduler be_sch;
          forwarding-class ef scheduler ef_sch;
          forwarding-class af scheduler af_sch;
          forwarding-class nc scheduler nc_sch;
          forwarding-class video scheduler video_sch;
          forwarding-class data scheduler data_sch;
        }
        be_ef_smap {
          forwarding-class be scheduler be_sch;
          forwarding-class ef scheduler ef_sch;
        }
        af_smap {
          forwarding-class af scheduler af_sch;
        }
        be_ef_af_nc_smap {
          forwarding-class be scheduler be_sch;
          forwarding-class ef scheduler ef_sch;
          forwarding-class af scheduler af_sch;
          forwarding-class nc scheduler nc_sch;
        }
        voice_video_game_data_smap {
          forwarding-class voice scheduler voice_sch;
          forwarding-class video scheduler video_sch;
          forwarding-class game scheduler game_sch;
          forwarding-class data scheduler data_sch;
        }
      }
      schedulers {
        "$junos-cos-scheduler" {
          transmit-rate percent "$junos-cos-scheduler-tx";
          buffer-size percent "$junos-cos-scheduler-bs";
          priority "$junos-cos-scheduler-pri";
          drop-profile-map loss-priority low protocol any drop-profile
            "$junos-cos-scheduler-dropfile-low";
          drop-profile-map loss-priority medium-low protocol any drop-profile
            "$junos-cos-scheduler-dropfile-medium-low";
          drop-profile-map loss-priority medium-high protocol any drop-profile
            "$junos-cos-scheduler-dropfile-medium-high";
          drop-profile-map loss-priority high protocol any drop-profile
            "$junos-cos-scheduler-dropfile-high";
        }
      }
    }
  }
  service {
```

```
variables {
    fc_1 default-value be;
    sch_1 default-value be_sch;
    sch-tx_1 default-value 20000000;
    sch-bs_1 default-value 10;
    sch-pri_1 default-value high;
    sch-drop-low_1 default-value d3;
    sch-drop-med-low_1 default-value d2;
    sch-drop-med-high_1 default-value d1;
    sch-drop-high_1 default-value d0;
    sch-drop-any_1 default-value d3;
    fc_2 default-value af;
    sch_2 default-value af_sch;
    sch-tx_2 default-value 10;
    sch-bs_2 default-value 10;
    sch-pri_2 default-value high;
    sch-drop-low_2 default-value d3;
    sch-drop-med-low_2 default-value d2;
    sch-drop-med-high_2 default-value d1;
    sch-drop-high_2 default-value d0;
    sch-drop-any_2 default-value d3;
    fc_3 default-value voice;
    sch_3 default-value voice_sch;
    sch-tx_3 default-value 20000000;
    sch-bs_3 default-value 10;
    sch-pri_3 default-value high;
    sch-drop-low_3 default-value d3;
    sch-drop-med-low_3 default-value d2;
    sch-drop-med-high_3 default-value d1;
    sch-drop-high_3 default-value d0;
    sch-drop-any_3 default-value d3;
    fc_4 default-value game;
    sch_4 default-value game_sch;
    sch-tx_4 default-value 10;
    sch-bs_4 default-value 10;
    sch-pri_4 default-value high;
    sch-drop-low_4 default-value d3;
    sch-drop-med-low_4 default-value d2;
    sch-drop-med-high_4 default-value d1;
    sch-drop-high_4 default-value d0;
    sch-drop-any_4 default-value d3;
    scheduler-map default-value all_smap;
}
class-of-service {
    scheduler-maps {
        "$scheduler-map" {
            forwarding-class "$fc_1" scheduler "$sch_1";
            forwarding-class "$fc_2" scheduler "$sch_2";
            forwarding-class "$fc_3" scheduler "$sch_3";
            forwarding-class "$fc_4" scheduler "$sch_4";
        }
    }
    schedulers {
        "$sch_1" {
            transmit-rate "$sch-tx_1";
            buffer-size percent "$sch-bs_1";
```

```
            priority "$sch-pri_1";
            drop-profile-map loss-priority low protocol any drop-profile
              "$sch-drop-low_1";
            drop-profile-map loss-priority medium-low protocol any drop-profile
              "$sch-drop-med-low_1";
            drop-profile-map loss-priority medium-high protocol any drop-profile
              "$sch-drop-med-high_1";
            drop-profile-map loss-priority high protocol any drop-profile
              "$sch-drop-high_1";
          }
          "$sch_2" {
            transmit-rate percent "$sch-tx_2";
            buffer-size percent "$sch-bs_2";
            priority "$sch-pri_2";
            drop-profile-map loss-priority low protocol any drop-profile
              "$sch-drop-low_2";
            drop-profile-map loss-priority medium-low protocol any drop-profile
              "$sch-drop-med-low_2";
            drop-profile-map loss-priority medium-high protocol any drop-profile
              "$sch-drop-med-high_2";
            drop-profile-map loss-priority high protocol any drop-profile
              "$sch-drop-high_2";
          }
          "$sch_3" {
            transmit-rate "$sch-tx_3";
            buffer-size percent "$sch-bs_3";
            priority "$sch-pri_3";
            drop-profile-map loss-priority low protocol any drop-profile
              "$sch-drop-low_3";
            drop-profile-map loss-priority medium-low protocol any drop-profile
              "$sch-drop-med-low_3";
            drop-profile-map loss-priority medium-high protocol any drop-profile
              "$sch-drop-med-high_3";
            drop-profile-map loss-priority high protocol any drop-profile
              "$sch-drop-high_3";
          }
          "$sch_4" {
            transmit-rate percent "$sch-tx_4";
            buffer-size percent "$sch-bs_4";
            priority "$sch-pri_4";
            drop-profile-map loss-priority low protocol any drop-profile
              "$sch-drop-low_4";
            drop-profile-map loss-priority medium-low protocol any drop-profile
              "$sch-drop-med-low_4";
            drop-profile-map loss-priority medium-high protocol any drop-profile
              "$sch-drop-med-high_4";
            drop-profile-map loss-priority high protocol any drop-profile
              "$sch-drop-high_4";
          }
        }
      }
    }
    service_2 {
      variables {
        fc_1 default-value be;
        sch_1 default-value be_sch;
```

```
        sch-tx_1 default-value 10;
        sch-bs_1 default-value 10;
        sch-pri_1 default-value high;
        sch-drop-low_1 default-value d3;
        sch-drop-med-low_1 default-value d2;
        sch-drop-med-high_1 default-value d1;
        sch-drop-high_1 default-value d0;
        sch-drop-any_1 default-value d3;
        scheduler-map default-value all_smap;
      }
      class-of-service {
        scheduler-maps {
          "$scheduler-map" {
            forwarding-class "$fc_1" scheduler "$sch_1";
          }
        }
        schedulers {
          "$sch_1" {
            transmit-rate percent "$sch-tx_1";
            buffer-size percent "$sch-bs_1";
            priority "$sch-pri_1";
            drop-profile-map loss-priority low protocol any drop-profile
              "$sch-drop-low_1";
            drop-profile-map loss-priority medium-low protocol any drop-profile
              "$sch-drop-med-low_1";
            drop-profile-map loss-priority medium-high protocol any drop-profile
              "$sch-drop-med-high_1";
            drop-profile-map loss-priority high protocol any drop-profile
              "$sch-drop-high_1";
          }
        }
      }
    }
  }
```

4. The network administrator configures DHCP and RADIUS to grant access and services to the interfaces referenced by the **subscriber** dynamic profile.

```
[edit]
  forwarding-options {
  dhcp-relay {
    traceoptions {
      file size 1g;
      flag all;
    }
    dynamic-profile subscriber aggregate-clients replace;
    server-group {
      subscriber-server {
        3.1.1.2;
      }
    }
    active-server-group subscriber-server;
    group relay-0 {
      authentication {
        password pwd0;
        username-include {
```

```
                user-prefix user0;
                mac-address;
              }
            }
            interface ge-1/1/0.100;
            interface ge-1/1/0.200;
          }
        }
      }
      radius-server {
        121.0.0.11 secret "$9$mPF/u0Icrv1RvL7V4oik.Pz3/CtOIE"; ## SECRET-DATA
      }
        profile subscriber-profile {
          authentication-order radius;
          radius {
            authentication-server 121.0.0.11;
            accounting-server 121.0.0.11;
          }
          radius-server {
            121.0.0.11 secret "$9$.mz6pu1hyKBIK8xdg4jHqmQF69A01R"; ## SECRET-DATA
          }
          accounting {
            order radius;
            statistics time;
          }
        }
```

Related Documentation

- Configuring Per-Unit Scheduling in a Dynamic Profile for Subscriber Access on page 53

Example: Providing Unique Rate Configurations for Schedulers in a Dynamic Profile

Combining static and dynamic schedulers in a dynamic profile enables you to provide subscribers with services that have unique scheduler definitions.

In this example, the network administrator configures the data service with a **transmit-rate** that is rate controlled using the **$junos-cos-scheduler-tx** predefined variable. RADIUS dynamically supplies the percentage value for the transmission rate that is specified in the RADIUS VSA to the data scheduler when the subscriber logs in.

For the best-effort service, the network administrator assigns the remaining transmission rate that is available.

```
schedulers {
  data-scheduler {
    transmit-rate percent rate-limit $junos-cos-scheduler-tx;
    buffer-size percent $junos-cos-scheduler-bs;
    priority $junos-cos-scheduler-pri;
    drop-profile-map loss-priority low protocol any drop-profile d0;
    drop-profile-map loss-priority medium-low protocol any drop-profile d1;
    drop-profile-map loss-priority medium-high protocol any drop-profile d2;
    drop-profile-map loss-priority high protocol any drop-profile d3;
    drop-profile-map loss-priority any protocol any drop-profile all;
  }
```

```
best-effort-scheduler {
  transmit-rate remainder;
  buffer-size percent $junos-cos-scheduler-bs;
  priority medium-high;
  drop-profile-map loss-priority low protocol any drop-profile
    $junos-cos-scheduler-dropfile-low;
  drop-profile-map loss-priority medium-low protocol any drop-profile d1;
  drop-profile-map loss-priority medium-high protocol any drop-profile
    $junos-cos-scheduler-dropfile-medium-high;
  drop-profile-map loss-priority high protocol any drop-profile d3;
  drop-profile-map loss-priority any protocol any drop-profile
    $junos-cos-scheduler-dropfile-any;
}
```

Related Documentation

- Configuring a Combination of Static and Dynamic Scheduler Parameters in a Scheduler Definition on page 62

Example: Configuring Initial CoS Parameters Dynamically Obtained from RADIUS

The following configuration is an example of a client dynamic profile in which initial CoS parameters are dynamically obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is applied.

For this example, assume that the RADIUS authentication server has been configured with traffic-shaping parameters (at Juniper Networks VSA 26-108) and CoS scheduling and queuing parameters (at Juniper Networks VSA 26–146).

The subscriber interface is a single-unit static gigabit Ethernet VLAN interface on an EQ DPC port:

```
[edit]
interfaces {
  ge-9/0/3 {
    hierarchical-scheduler;
    vlan-tagging;
    unit 100 {
      vlan-id 100;
      family inet {
        address 192.168.32.2/24;
      }
    }
  }
}
```

The client dynamic profile **residential_silver** attaches the traffic-control profile **tcp_1** to the subscriber interface that is defined in the dynamic profile using the **$junos-interface-ifd-name** predefined variable.

```
[edit]
dynamic-profiles {
  residential_silver {
    interfaces {
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit" {
          family inet;
```

```
        }
      }
    }
    class-of-service {
      interfaces {
        “$junos-interface-ifd-name” {
          unit “$junos-underlying-interface-unit” {
            output-traffic-control-profile tcp_1;
          }
        }
      }
    }
  }
}
```

The traffic-control profile **tcp_1**, references Junos OS predefined variables to obtain a scheduler-map name and traffic-shaping parameter values from RADIUS when a subscriber logs in. For this example, assume that the RADIUS server replaces the Junos OS predefined variable **$junos-cos-scheduler-map** scheduler-map name **business_smap_1**. The scheduler map **business_smap_1** is configured in the client dynamic profile:

```
[edit]
dynamic-profiles {
  residential_silver {
    class-of-service {
      traffic-control-profiles {
        tcp_1 {
          scheduler-map “$junos-cos-scheduler-map”; # ’business_smap_1’
          shaping-rate "$junos-cos-shaping-rate";
          guaranteed-rate "$junos-cos-guaranteed-rate";
          delay-buffer-rate "$junos-cos-delay-buffer-rate";
        }
      }
      scheduler-maps {
        business_smap_1 {
          forwarding-class best-effort scheduler be_sched;
          forwarding-class ef scheduler home_sched
        }
      }
    }
  }
}
```

A scheduler definition references Junos OS predefined variables to obtain scheduler configurations from RADIUS when a subscriber logs in. For this example, assume that the RADIUS server provides scheduler configurations for schedulers named **be_sched** and **home_sched**, which are included in the scheduler map **business_smap_1**:

```
[edit]
dynamic-profiles {
  residential_silver {
    class-of-service {
      schedulers {
        “$junos-cos-scheduler” { # ’be_sched’ and ’home_sched’
          transmit-rate "$junos-cos-scheduler-tx";
          buffer-size "$junos-cos-scheduler-bs";
          priority "$junos-cos-scheduler-pri";
```



```
            drop-profile-map loss-priority low protocol any drop-profile
              "$junos-cos-scheduler-dropfile-low";
            drop-profile-map loss-priority medium-low protocol any drop-profile
              "$junos-cos-scheduler-dropfile-medium-low";
            drop-profile-map loss-priority medium-high protocol any drop-profile
              "$junos-cos-scheduler-dropfile-medium-high";
            drop-profile-map loss-priority high protocol any drop-profile
              "$junos-cos-scheduler-dropfile-high";
          }
        }
      }
    }
  }
```

Static configurations for CoS consist of configurations for the forwarding classes used in the scheduler map **business_smap_1** and configurations for drop-profile names provided by RADIUS for as part of the scheduler configurations provided (for **be_sched** and **home_sched**) when a subscriber logs in:

```
[edit]
  class-of-service {
    forwarding-classes {
      queue 0 best-effort;
      queue 1 ef;
    }
    drop-profiles {
      . . . configurations_for_drop_profile_names_provided_by_RADIUS . . .
    }
  }
}
```

Related Documentation

- *Subscriber Activation and Service Management in an Access Network*
- *Dynamic Profiles Overview*
- *Dynamic Variables Overview*
- *Junos OS Predefined Variables*
- Subscriber Interfaces That Provide Initial CoS Parameters Dynamically Obtained from RADIUS on page 21
- Configuring Initial CoS Parameters Dynamically Obtained from RADIUS on page 67

Example: Configuring Aggregate Scheduling of Queues for Residential Subscribers on Static IP Demux Interfaces

In this example, scheduling is configured for a residential subscriber. Each forwarding class represents a multiplay service (voice, video, and data), and is equivalent to a queue.

An interface set of IP demux interfaces represents a DSLAM, and provides shaping of subscribers services to a DSLAM aggregate rate.

```
[edit]
interfaces {
  interface-set demux-set {
```

```
    interface demux0 {
      unit 0;
      unit 1;
    }
  }
  ge-2/0/1 {
    vlan-tagging;
    unit 1 {
      per-session-scheduler;
      vlan-id 1;
      demux-source inet;
      family inet {
        address 4.4.4.4/24;
      }
    }
  }
  demux0 {
    unit 0 {
      demux-options {
        underlying-interface ge-2/0/1.1;
      }
      family inet {
        address 1.1.1.1/24;
        demux-source {
          1.1.1.0/24;
        }
      }
    }
    unit 1 {
      demux-options {
        underlying-interface ge-2/0/1.1;
      }
      family inet {
        address 1.1.2.1/24;
        demux-source {
          1.1.2.0/24;
        }
      }
    }
  }
}
class-of-service {
  traffic-control-profiles {
    T1 {
      scheduler-map m1;
      shaping-rate 5m;
    }
    T2 {
      shaping-rate 60m;
    }
  }
  interfaces {
    interface-set demux-set {
      output-traffic-control-profile T2;
    }
    demux0 {
```

```
        unit 0 {
          output-traffic-control-profile T1;
        }
        unit 1 {
          output-traffic-control-profile T1;
        }
      }
    }
    scheduler-maps {
      m1 {
        forwarding-class best-effort scheduler s0;
        forwarding-class expedited-forwarding scheduler s1;
        forwarding-class assured-forwarding scheduler s2;
        forwarding-class network-control scheduler s3;
      }
    }
    schedulers {
      s0 {
        transmit-rate percent 10;
        buffer-size percent 10;
      }
      s1 {
        transmit-rate percent 20;
        buffer-size percent 20;
      }
      s2 {
        transmit-rate percent 30;
        buffer-size percent 30;
      }
      s3 {
        transmit-rate percent 40;
        buffer-size percent 40;
      }
    }
  }
```

Related Documentation

- *CoS and Static IP Demux Interface Set Overview*
- *Configuring Static Subscriber Interfaces Using IP Demux Interfaces*

Example: Configuring Hierarchical Scheduling and Queuing for a Static PPPoE Subscriber Interface

In this example, the network administrator defines hierarchical queuing and scheduler parameters by configuring traffic-control profile and binding it directly to a PPPoE subscriber interface.

This configuration is supported on the IQ2E PIC.

To use this configuration in a broadband access network, each forwarding class can represent one type of services provided to a household customer and is mapped to a queue. Each PPPoE interface represents a household and provides shaping of all household traffic to an aggregate rate. All of the PPPoE interfaces on the physical interfaces are shaped to the underlying physical interface rate.

Table 26 on page 124 lists the scheduler and queue mapping for this configuration.

Table 26: Scheduler Per Logical Interface Mapping

Level	Type	Mapping
4	Queue	PPPoE interface
3	Scheduler	PPPoE interface
2	Scheduler	—
1	Scheduler	Underlying physical interface

```
interfaces {
   ge-3/0/3 {
      hierarchical-scheduler;
      vlan-tagging;
      unit 0 {
         encapsulation ppp-over-ether;
         vlan-id 100;
      }
   }
   pp0 {
      unit 0 {
         pppoe-options {
            underlying-interface ge-3/0/3.0;
            server;
         }
         family inet {
            address 120.20.20.20/32 {
               destination 120.20.20.21;
            }
         }
      }
      unit 1 {
         pppoe-options {
            underlying-interface ge-3/0/3.0;
            server;
         }
         family inet {
            address 130.30.30.30/32 {
               destination 130.30.30.31;
            }
         }
      }
      unit 2 {
         pppoe-options {
            underlying-interface ge-3/0/3.0;
            server;
         }
         family inet {
            address 140.40.40.40/32 {
               destination 140.40.40.41;
```

```
        }
      }
    }
  }
}
  class-of-service {
    traffic-control-profiles {
      tcp {
        scheduler-map data_smap;
        shaping-rate 50k;
        guaranteed-rate 10k;
      }
    }
    interfaces {
      pp0 {
        unit 0 {
          output-traffic-control-profile tcp;
        }
          unit 1 {
            output-traffic-control-profile tcp;
            unit 2 {
              output-traffic-control-profile tcp;
              }
              }
            }
            forwarding-classes {
              queue 0 be;
              queue 1 ef;
              queue 3 nc;
              queue 2 af;
            }
            scheduler-maps {
              data_smap {
                forwarding-class be scheduler be_sch;
              }
              voice_data_smap {
                forwarding-class be scheduler be_sch;
              }
              vid_data_smap {
                forwarding-class ef scheduler ef_sch;
              }
            }
            schedulers {
              be_sch {
                transmit-rate percent 10;
                buffer-size remainder;
                priority low;
              }
              ef_sch {
                transmit-rate percent 10;
                buffer-size remainder;
                priority low;
              }
              af_sch {
                transmit-rate percent 10;
                buffer-size remainder;
```

```
        priority low;
      }
      nc_sch {
        transmit-rate percent 10;
        buffer-size remainder;
        priority low;
      }
    }
```

Related Documentation

- CoS for PPPoE Subscriber Interfaces Overview on page 12
- Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface on page 87

Example: Configuring Hierarchical Scheduling and Queuing for an Underlying Static PPPoE Subscriber Interface

In this example, the network administrator defines hierarchical queues and scheduler parameters by configuring a traffic-control profile and binding it directly to a PPPoE subscriber interface. The network administrator then configures the traffic-control profile on the underlying interface where a group of PPPoE interfaces reside.

This configuration is supported on the IQ2E PIC.

To use this configuration in a broadband access network, each forwarding class represents one type of services provided to a household customer and is mapped to a queue. Each PPPoE interface represents a household and provides shaping of all household traffic to an aggregate rate. The underlying logical interface where a group of PPPoE interfaces resides represents a DSLAM and provides shaping to the DSLAM rate.

Table 27 on page 126 lists the scheduler and queue mapping for this configuration.

Table 27: Scheduler per Underlying Interface Mapping

Level	Type	Mapping
4	Queue	PPPoE interface
3	Scheduler	PPPoE interface
2	Scheduler	Underlying logical interface
1	Scheduler	Underlying interface

```
interfaces {
  ge-3/0/3 {
    hierarchical-scheduler;
    vlan-tagging;
    unit 0 {
      encapsulation ppp-over-ether;
      vlan-id 100;
    }
    unit 1 {
```

```
            vlan-id 101;
        }
    }
    pp0 {
        hierarchical-scheduler;
        unit 0 {
            pppoe-options {
                underlying-interface ge-3/0/3.0;
                server;
            }
            family inet {
                address 120.20.20.20/32 {
                    destination 120.20.20.21;
                }
            }
        }
        unit 1 {
            pppoe-options {
                underlying-interface ge-3/0/3.0;
                server;
            }
            family inet {
                address 130.30.30.30/32 {
                    destination 130.30.30.31;
                }
            }
        }
        unit 2 {
            pppoe-options {
                underlying-interface ge-3/0/3.0;
                server;
            }
            family inet {
                address 140.40.40.40/32 {
                    destination 140.40.40.41;
                }
            }
        }
    }
}
    class-of-service {
        traffic-control-profiles {
            tcp1 {
                scheduler-map data_smap;
                shaping-rate 50k;
                guaranteed-rate 10k;
            }
            tcp2 {
                scheduler-map data_smap;
                shaping-rate 50m;
                guaranteed-rate 10m;
            }
        }
        interfaces {
            pp0 {
                unit 0 {
```

```
    output-traffic-control-profile tcp1;
}
  unit 1 {
    output-traffic-control-profile tcp1;
    }
    unit 2 {
      output-traffic-control-profile tcp1;
      }
      }
      ge-3/0/3 {
        unit 0 {
          output-traffic-control-profile tcp2;
        }
      }
    }
    ...
}
```

Related Documentation

- CoS for PPPoE Subscriber Interfaces Overview on page 12
- Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface on page 87
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4

Example: Configuring Hierarchical Scheduling and Queuing for an Interface Set of Static PPPoE Subscriber Interfaces

In this example, the network administrator defines hierarchical queues and scheduler parameters by configuring traffic-control profile and binding it directly to a PPPoE subscriber interface. The network administrator then configures the traffic-control profile on a set of PPPoE interfaces.

This configuration is supported on the IQ2E PIC.

To use this configuration in a broadband access network, each forwarding class represents one type of services provided to a household customer and is mapped to a queue. Each PPPoE interface represents a household and provides shaping of all household traffic to an aggregate rate. In addition, the PPPoE interface-set configuration provides shaping of traffic for a group of PPPoE interface on a DSLAM to a DSLAM aggregate rate.

Table 28 on page 128 lists the scheduler and queue mapping for this configuration.

Table 28: Scheduler per Logical Interface with Interface Set Mapping

Level	Type	Mapping
4	Queue	PPPoE interface
3	Scheduler	PPPoE interface
2	Scheduler	Set of PPPoE interfaces

Table 28: Scheduler per Logical Interface with Interface Set Mapping *(continued)*

Level	Type	Mapping
1	Scheduler	Underlying physical interface

```
interfaces {
   interface-set iflset1 {
      interface pp0 {
         unit 0;
         unit 1;
         unit 2;
         }
      }
      pp0 {
         unit 0 {
            pppoe-options {
               underlying-interface ge-3/0/3.0;
               server;
            }
            family inet {
               address 120.20.20.20/32 {
                  destination 120.20.20.21;
               }
            }
         }
         unit 1 {
            pppoe-options {
               underlying-interface ge-3/0/3.0;
               server;
            }
            family inet {
               address 130.30.30.30/32 {
                  destination 130.30.30.31;
               }
            }
         }
         unit 2 {
            pppoe-options {
               underlying-interface ge-3/0/3.0;
               server;
            }
            family inet {
               address 140.40.40.40/32 {
                  destination 140.40.40.41;
               }
            }
         }
      }
      ge-3/0/3 {
         hierarchical-scheduler;
         vlan-tagging;
         unit 0 {
            encapsulation ppp-over-ether;
```

```
        vlan-id 100;
      }
      unit 1 {
        vlan-id 101;
      }
      unit 2 {
        vlan-id 102;
      }
    }
  }
  class-of-service {
    traffic-control-profiles {
      tcp1 {
        scheduler-map data_smap;
        shaping-rate 50k;
        guaranteed-rate 10k;
      }
      tcp2 {
        scheduler-map data_smap;
        shaping-rate 50m;
        guaranteed-rate 10m;
      }
    }
    interfaces {
      pp0 {
        unit 0 {
          output-traffic-control-profile tcp1;
        }
        unit 1 {
          output-traffic-control-profile tcp1;
        }
        unit 2 {
          output-traffic-control-profile tcp1;
        }
        interface-set iflset1 {
          output-traffic-control-profile tcp2;
        }
        ...
      }
```

Related Documentation

- CoS for PPPoE Subscriber Interfaces Overview on page 12
- Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface on page 87
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4

Example: Configuring a Dynamic Interface Set of VLAN Subscribers

- Requirements on page 131
- Overview on page 131
- Configuring the Dynamic VLANs on page 131
- Configuring Dynamic Traffic Scheduling and Shaping on page 133

Requirements

This example uses the following software and hardware components:

- Junos OS Release 10.4
- MX Series Router with MPCs

Overview

In this example, the network administrator groups dynamic VLAN interfaces in an interface set. The interface set is configured in a dynamic profile, and enables hierarchical scheduling for the VLAN interfaces for a multiplay service.

DHCP is used as the access method, and RADIUS is used as the authentication method for the interfaces associated with the interface set.

Configuring the Dynamic VLANs

CLI Quick Configuration

To quickly configure the dynamic VLANs, copy the following commands and paste them into the router terminal window:

```
[edit]
edit dynamic-profiles vlan-prof
edit interfaces $junos-interface-ifd-name unit $junos-interface-unit
set vlan-id $junos-vlan-id
set demux-source inet
set family inet unnumbered-address lo0.0 preferred-source-address 100.20.32.2
top
edit interfaces ge-1/0/0
set hierarchical-scheduler
set vlan-tagging
edit auto-configure vlan-ranges dynamic-profile vlan-prof
set ranges any
set accept inet
top
set interfaces lo0 unit 0 family inet address 100.20.32.2/32
```

Configuring the Dynamic Profile for the Autoconfigured VLANs

Step-by-Step Procedure

In this section, you create a dynamic profile for the VLAN IDs to be automatically assigned when subscribers log in.

To configure the dynamic profile for the VLANs:

1. Configure the dynamic profile.

   ```
   [edit]
   user@host#edit dynamic-profile vlan-prof
   ```

2. Configure the interfaces.

```
[edit dynamic-profiles vlan-prof]
user@host#edit interfaces $junos-interface-ifd-name unit $junos-interface-unit
```

3. Add the VLAN ID variable.

```
[edit dynamic-profiles vlan-prof interfaces $junos-interface-ifd-name unit
  $junos-interface-unit]
user@host#set vlan-id $junos-vlan-id
```

4. Configure the demux source as IPv4.

```
[edit dynamic-profiles vlan-prof interfaces $junos-interface-ifd-name unit
  $junos-interface-unit]
user@host#set demux-source inet
```

5. Configure the family.

```
[edit dynamic-profiles vlan-prof interfaces $junos-interface-ifd-name unit
  $junos-interface-unit]
user@host#set family inet unnumbered-address lo0.0 preferred-source-address
  100.20.32.2
```

Configuring the VLAN Interfaces

Step-by-Step Procedure

To configure the VLAN interfaces:

1. Create the VLAN interface.

```
[edit]
user@host# edit interfaces ge-1/0/0
```

2. Enable hierarchical scheduling.

```
[edit interfaces ge-1/0/0]
user@host# set hierarchical-scheduler
```

3. Configure VLAN tagging.

```
[edit interfaces ge-1/0/0]
user@host# set vlan-tagging
```

4. Configure auto-configuration for the dynamic profile.

```
[edit interfaces ge-1/0/0]
user@host# edit auto-configure vlan-ranges dynamic-profile vlan-prof
```

5. Configure any VLAN ID range.

```
[edit interfaces ge-1/0/0 auto-configure vlan-ranges dynamic-profile vlan-prof]
user@host# set ranges any
```

6. Specify IPv4 traffic for the VLAN.

```
[edit interfaces ge-1/0/0 auto-configure vlan-ranges dynamic-profile vlan-prof]
user@host# set accept inet
```

Configuring the Loopback Interface

Step-by-Step Procedure

To configure the loopback interface:

1. Create the loopback interface.

```
[edit]
user@host# edit interfaces lo0
```

2. Configure the unit and the family.

```
[edit intefaces lo0]
user@host# set unit 0 family inet address 100.20.32.2/32
```

Configuring Dynamic Traffic Scheduling and Shaping

CLI Quick Configuration

To quickly configure the traffic scheduling and shaping parameters, copy the following commands and paste them into the router terminal window:

```
[edit]
edit dynamic-profiles multiplay class-of-service schedulers be_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit ef_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit af_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit nc_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit voice_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit video_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit game_sch
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up
edit data_sch
```

```
set transmit-rate percent 12
set buffer-size percent 12
set priority low
up 2
edit scheduler-maps all_smap
set forwarding-class be scheduler be_sch
set forwarding-class ef scheduler ef_sch
set forwarding-class af scheduler af_sch
set forwarding-class nc scheduler nc_sch
set forwarding-class voice scheduler voice_sch
set forwarding-class video scheduler video_sch
set forwarding-class game scheduler game_sch
set forwarding-class data scheduler data_sch
up 2
edit traffic-control-profiles multiplay
set scheduler-map all_smap
set shaping-rate 100m
set guaranteed-rate 20m
```

Configuring the Schedulers in the Dynamic Profile

Step-by-Step Procedure

In this section, you create a dynamic profile for the multiplay service and configure scheduling and shaping.

To configure the schedulers:

1. Create the **multiplay** dynamic profile.

   ```
   [edit]
   user@host# edit dynamic-profiles multiplay class-of-service schedulers
   ```

2. Configure the best effort scheduler.

   ```
   [edit dynamic-profiles multiplay class-of-service schedulers]
   user@host# edit be_sch
   user@host# set transmit-rate percent 12
   user@host# set buffer-size percent 12
   user@host# set priority low
   ```

3. Configure the expedited forwarding scheduler.

   ```
   [edit dynamic-profiles multiplay class-of-service schedulers]
   user@host# edit ef_sch
   user@host# set transmit-rate percent 12
   user@host# set buffer-size percent 12
   user@host# set priority low
   ```

4. Configure the assured forwarding scheduler.

   ```
   [edit dynamic-profiles multiplay class-of-service schedulers]
   user@host# edit af_sch
   user@host# set transmit-rate percent 12
   user@host# set buffer-size percent 12
   user@host# set priority low
   ```

5. Configure the network control scheduler.

   ```
   [edit dynamic-profiles multiplay class-of-service schedulers]
   user@host# edit nc_sch
   ```

```
user@host# set transmit-rate percent 12
user@host# set buffer-size percent 12
user@host# set priority low
```

6. Configure the voice scheduler.

```
[edit dynamic-profiles multiplay class-of-service schedulers]
user@host# edit voice_sch
user@host# set transmit-rate percent 12
user@host# set buffer-size percent 12
user@host# set priority low
```

7. Configure the video scheduler.

```
[edit dynamic-profiles multiplay class-of-service schedulers]
user@host# edit video_sch
user@host# set transmit-rate percent 12
user@host# set buffer-size percent 12
user@host# set priority low
```

8. Configure the gaming scheduler.

```
[edit dynamic-profiles multiplay class-of-service schedulers]
user@host# edit game_sch
user@host# set transmit-rate percent 12
user@host# set buffer-size percent 12
user@host# set priority low
```

9. Configure the data scheduler.

```
[edit dynamic-profiles multiplay class-of-service schedulers]
user@host# edit data_sch
user@host# set transmit-rate percent 12
user@host# set buffer-size percent 12
user@host# set priority low
```

Configuring the Scheduler Map in the Dynamic Profile

Step-by-Step Procedure

To configure the scheduler map:

1. Configure the scheduler map for all of the services.

```
[edit dynamic-profiles multiplay class-of-service]
user@host# edit scheduler-maps all_smap
```

2. Configure the forwarding classes for each service in the scheduler map.

```
[edit dynamic-profiles multiplay class-of-service scheduler-maps all_smap]
user@host# set forwarding-class be scheduler be_sch
user@host# set forwarding-class ef scheduler ef_sch
user@host# set forwarding-class af scheduler af_sch
user@host# set forwarding-class nc scheduler nc_sch
user@host# set forwarding-class voice scheduler voice_sch
user@host# set forwarding-class video scheduler video_sch
user@host# set forwarding-class game scheduler game_sch
user@host# set forwarding-class data scheduler data_sch
```

Configuring the Traffic-Control Profile in the Dynamic Profile

Step-by-Step Procedure

To configure the traffic-control profile the interface set:

1. Configure the traffic-control profile.

   ```
   [edit dynamic-profiles multiplay class-of-service]
   user@host# edit traffic control-profiles multiplay
   ```

2. Configure the scheduler map.

   ```
   [edit dynamic-profiles multiplay class-of-service traffic control-profiles multiplay]
   user@host# set scheduler-map all_smap
   ```

3. Configure the shaping rate.

   ```
   [edit dynamic-profiles multiplay class-of-service traffic control-profiles multiplay]
   user@host# set shaping-rate 100m
   ```

4. Configure the guaranteed rate.

   ```
   [edit dynamic-profiles multiplay class-of-service traffic control-profiles multiplay]
   user@host# set guaranteed-rate 20m
   ```

Configuring the Interface Set in the Dynamic Profile

CLI Quick Configuration

To quickly configure the interface set, copy the following commands and paste them into the router terminal window:

```
[edit]
edit dynamic-profiles multiplay
edit interfaces interface-set $junos-interface-set-name
set interface $junos-interface-ifd-name unit $junos-underlying-interface-unit
top
edit class-of-service interfaces interface-set
set output-traffic-control-profile multiplay
```

Configuring the Interfaces for the Interface Set

Step-by-Step Procedure

To configure the interface variable for the interface set:

1. Configure the dynamic profile for the interface set.

   ```
   [edit]
   user@host#edit dynamic-profiles multiplay
   ```

2. Configure the interface using the Junos OS predefined variable.

   ```
   [edit dynamic-profiles multiplay]
   user@host#edit interfaces $junos-interface-ifd-name unit
     $junos-underlying-interface-unit
   ```

3. Configure the family.

   ```
   [edit dynamic-profiles multiplay interfaces $junos-interface-set-name unit
     $junos-underlying-interface-unit]
   user@host#set family inet unnumbered-address lo0.0 preferred-source-address
     100.20.32.2
   ```

Configuring the Interface Set

Step-by-Step Procedure

To configure the interface set:

1. Configure the interface set using the Junos OS predefined variable.

```
[edit dynamic-profiles multiplay]
user@host#edit interfaces interface-set $junos-interface-set-name
```

2. Add the dynamic VLAN interfaces to the interface set.

```
[edit dynamic-profiles multiplay interfaces $junos-interface-set-name]
user@host#set interface $junos-interface-ifd-name unit
  $junos-underlying-interface-unit
```

Applying the Traffic-Control Profile to the Interface Set

Step-by-Step Procedure

You apply the traffic-control profile outside of the dynamic profile in the **[edit class-of-service]** hierarchy.

To apply the traffic-control profile:

1. Specify the interface set to which you want to apply the traffic-control profile.

```
[edit class-of-service]
user@host#edit interfaces interface-set dynamic-set
```

2. Attach the output traffic-control profile defined in the dynamic profile to the interface set.

```
[edit class-of-service interfaces]
user@host#set output-traffic-control-profile multiplay
```

Configuring DHCP Access

CLI Quick Configuration

To quickly configure DHCP access, copy the following commands and paste them into the router terminal window:

```
[edit]
edit system services dhcp-local-server authentication
set password multiplay
set username-include user-prefix multiplay
up 1
set dynamic-profile dhcp-vlan-prof aggregate-clients replace
set group vlans interface ge-1/0/0
top
edit access address-assignment pool v4 family inet
set network 100.20.0.0/16
set range limited low 100.20.0.10
set range limited high 100.20.128.250
set dhcp-attributes maximum-lease-time 84600
```

Configuring the DHCP Local Server

Step-by-Step Procedure

To configure DHCP access:

1. Configure the DHCP local server.

   ```
   [edit system]
   user@host# edit services dhcp-local-server authentication
   ```

2. Set the password.

   ```
   [edit system services dhcp-local-server authentication]
   user@host# set password multiplay
   ```

3. Specify that you want to include optional information in the username.

   ```
   [edit system services dhcp-local-server authentication]
   user@host# set username-include user-prefix multiplay
   ```

4. Attach the dynamic profile with the interface set.

   ```
   [edit system services dhcp-local-server]
   user@host# set dynamic-profile dhcp-vlan-prof aggregate-clients replace
   ```

5. Configure a group for the VLAN interface.

   ```
   [edit system services dhcp-local-server]
   user@host# set group vlans interface ge-1/0/0
   ```

Configuring Address Assignment Pools

Step-by-Step Procedure

To configure address assignment pools:

1. Configure the pool of IPv4 addresses.

   ```
   [edit access]
   user@host#edit address-assignment pool v4 family inet
   ```

2. Configure the family of interfaces in the pool.

   ```
   [edit access address-assignment pool v4]
   user@host#set network 100.20.0.0/16
   ```

3. Configure the upper and lower bounds of the address range.

   ```
   [edit access address-assignment pool v4]
   user@host#set range limited low 100.20.0.10
   user@host#set range limited high 100.20.128.250
   ```

4. Configure the maximum length of time in seconds for which a subscriber can request and hold a lease.

   ```
   [edit access address-assignment pool v4]
   user@host#set dhcp-attributes maximum-lease-time 84600
   ```

Configuring RADIUS Authentication

CLI Quick Configuration

To quickly configure RADIUS authentication, copy the following commands and paste them into the router terminal window:

```
[edit]
edit access radius-server 172.28.30.108
set secret $9$1u5ErvW87bwgSr4Zji5T
set timeout 5
set retry 5
up 2
edit profile acc-prof
set authentication-order radius
set radius authentication-server 172.28.30.108
```

Configuring RADIUS Access

Step-by-Step Procedure

To configure RADIUS access:

1. Configure the RADIUS server.

   ```
   [edit access]
   user@host#edit radius-server 172.28.30.108
   ```

2. Configure the required secret (password) that the local router or switch passes to the RADIUS client.

   ```
   [edit access radius-server 172.28.30.108]
   user@host# set secret $9$1u5ErvW87bwgSr4Zji5T
   ```

3. Configure the length of time that the local router or switch waits to receive a response from a RADIUS server.

   ```
   [edit access radius-server 172.28.30.108]
   user@host# set timeout 5
   ```

4. Configure the number of times that the router or switch attempts to contact a RADIUS accounting server.

   ```
   [edit access radius-server 172.28.30.108]
   user@host# set retry 5
   ```

5. Configure the access profile.

   ```
   [edit access]
   user@host#edit profile acc-prof
   ```

6. Configure the authentication order.

   ```
   [edit access profile acc-prof ]
   user@host# set authentication-order radius
   ```

7. Configure the authentication server.

   ```
   [edit access profile acc-prof]
   user@host#set radius authentication-server 172.28.30.108
   ```

Results

```
dynamic-profiles {
  vlan-prof {
    interfaces {
      “$junos-interface-ifd-name” {
        unit "$junos-interface-unit" {
          vlan-id "$junos-vlan-id";
```

```
            demux-source inet;
            family inet {
              unnumbered-address lo0.0 preferred-source-address 100.20.32.2;
            }
          }
        }
      }
    }
    multiplay {
      class-of-service {
        traffic-control-profiles {
          multiplay {
            scheduler-map all_smap;
            shaping-rate 100m;
            guaranteed-rate 20m;
          }
        }
        interfaces {
            interface-set “$junos-interface-set-name” {
              interface “$junos-interface-ifd-name” {
                unit “$junos-underlying-interface-unit”;
              }
            }
          “$junos-interface-ifd-name” {
            unit "$junos-interface-unit" {
              output-traffic-control-profile multiplay;
            }
          }
        }
        scheduler-maps {
          all_smap {
            forwarding-class be scheduler be_sch;
            forwarding-class ef scheduler ef_sch;
            forwarding-class af scheduler af_sch;
            forwarding-class nc scheduler nc_sch;
            forwarding-class voice scheduler voice_sch;
            forwarding-class video scheduler video_sch;
            forwarding-class game scheduler game_sch;
            forwarding-class data scheduler data_sch;
          }
        }
        schedulers {
          be_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
          }
          ef_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
          }
          af_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
```

```
        }
        nc_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
        }
        voice_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
        }
        video_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
        }
        game_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
        }
        data_sch {
            transmit-rate percent 12;
            buffer-size percent 12;
            priority low;
        }
    }
    }
}
access {
    radius-server {
        172.28.30.108 {
            secret "$9$1u5ErvW87bwgSr4Zji5T"; ## SECRET-DATA
            timeout 5;
            retry 5;
        }
    }
    profile acc-prof {
        authentication-order radius;
        radius {
            authentication-server 172.28.30.108;
        }
    }
    address-assignment {
        pool v4 {
            family inet {
                network 100.20.0.0/16;
                range limited {
                    low 100.20.0.10;
                    high 100.20.128.250;
                }
                dhcp-attributes {
                    maximum-lease-time 84600;
                }
            }
        }
```

```
        }
      }
      class-of-service {
        interfaces {
          interface-set dynamic-set {
            output-traffic-control-profile multiplay;
          }
        }
      }
      interfaces {
        interface-set “$junos-interface-set-name” {
          interface "$junos-interface-ifd-name" {
            unit "$junos-underlying-interface-unit";
          }
        }
        "$junos-interface-ifd-name" {
          unit "$junos-underlying-interface-unit" {
            family inet {
              unnumbered-address lo0.0 preferred-source-address 100.20.32.2;
            }
          }
        }
      }
    }
  }
    interfaces {
      ge-1/0/0 {
        hierarchical-scheduler;
        vlan-tagging;
        auto-configure {
          vlan-ranges {
            dynamic-profile vlan-prof {
              accept inet;
              ranges {
                any;
              }
            }
          }
        }
      }
      lo0 {
        unit 0 {
          family inet {
            address 100.20.32.2/32;
          }
        }
      }
    }
    system {
      services {
        dhcp-local-server {
          authentication {
            password multiplay;
            username-include {
              user-prefix multiplay;
            }
```

```
        }
        dynamic-profile multiplay aggregate-clients replace;
        group vlans {
          interface ge-1/0/0.0;
        }
      }
    }
  }
```

Verification

To confirm that the configuration is correct, perform these tasks:

- Verifying the Interfaces that are Included in the Interface Set on page 143
- Verifying the Traffic Scheduling and Shaping Parameters for the Interface Set on page 143

Verifying the Interfaces that are Included in the Interface Set

Purpose Verify the interfaces included in the interface set.

Action `user@host>` **show interfaces interface-set dynamic-set terse**

Verifying the Traffic Scheduling and Shaping Parameters for the Interface Set

Purpose Verify that the traffic scheduling and shaping parameters are applied properly to an interface included in the interface set.

Action `user@host>` **show class-of-service interface**

Related Documentation

- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Configuring an Interface Set of Subscribers in a Dynamic Profile on page 85

Example: Configuring a Dynamic Service VLAN Interface Set of Subscribers in a Dynamic Profile

Interface sets enable you to provide hierarchical scheduling to a group of subscriber interfaces. In this example, by using the **$junos-svlan-interface-set-name** internal dynamic variable when specifying the interface set name, you can locally generate an interface set name for use by SVLAN interfaces based on the outer tag of the dual-tagged VLAN. The format of the generated variable is ***physical_interface_name - outer_VLAN_tag***.

- Requirements on page 144
- Overview on page 144
- Configuration on page 144
- Verification on page 146

Requirements

Dynamic SVLAN traffic shaping is supported only on Juniper Networks MX Series 3D Universal Edge Routers running Junos OS Release 11.4R2 or later.

Before you begin, configure the subscriber interfaces that you intend to include in the interface set. You can find general configuration instructions for the supported dynamic interface configuration in *Subscriber Interface Overview*and in the following:

- For dynamic VLAN interfaces, see *Configuring a Static or Dynamic VLAN Subscriber Interface over Aggregated Ethernet*.
- For dynamic IP demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using IP Demux Interfaces in Dynamic Profiles* and *Configuring a Static or Dynamic IP Demux Subscriber Interface over Aggregated Ethernet*.
- For dynamic VLAN demux interfaces, see *Configuring Dynamic Subscriber Interfaces Using VLAN Demux Interfaces in Dynamic Profiles*.

Overview

Interface sets enable you to provide hierarchical scheduling to a group of subscriber interfaces. By using the **$junos-svlan-interface-set-name** internal dynamic variable when specifying the interface set name, you can locally generate an interface set name for use by SVLAN interfaces based on the outer tag of the dual-tagged VLAN. The format of the generated variable is ***physical_interface_name - outer_VLAN_tag***.

This example includes the following statements:

- **interface-set**—Configures the name of the scheduler for dynamic CoS. In this example, you use the **$junos-svlan-interface-set-name** variable to obtain the locally generated interface set name for use by SVLAN interfaces based on the outer tag of the dual-tagged VLAN.
- **output-traffic-control-profile**—Applies an output traffic scheduling and shaping profile to the interface set.
- **output-traffic-control-profile-remaining**—Applies an output traffic scheduling and shaping profile for remaining traffic to the interface set.

Configuration

CLI Quick Configuration

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

```
[edit]
set dynamic-profiles profile-dhcp-ipdemux interfaces interface-set
   $junos-svlan-interface-set-name interface $junos-interface-ifd-name unit
   $junos-underlying-interface-unit
set dynamic-profiles profile-dhcp-ipdemux interfaces $junos-interface-ifd-name unit
   $junos-underlying-interface-unit
```

```
set class-of-service traffic-control-profiles tcp1 scheduler-map schedMap
set class-of-service traffic-control-profiles tcp1 shaping-rate 50m
set class-of-service traffic-control-profiles tcp1 guaranteed-rate 200k
set class-of-service traffic-control-profiles tcp3 scheduler-map ss1q0q1
set class-of-service traffic-control-profiles tcp3 shaping-rate 20m
set class-of-service traffic-control-profiles tcp3 guaranteed-rate 5m
set class-of-service interfaces interface-set ae0-111 output-traffic-control-profile tcp1
set class-of-service interfaces interface-set ae0-111
  output-traffic-control-profile-remaining tcp3
```

Step-by-Step Procedure

To configure an SVLAN interface set of subscriber interfaces:

1. Access the dynamic profile you want to modify for interface sets.

   ```
   [edit]
   user@host# edit dynamic-profiles profile-dhcp-ipdemux
   ```

2. Access the dynamic profile interface configuration.

   ```
   [edit dynamic-profiles profile-dhcp-ipdemux]
   user@host# edit interfaces
   ```

3. Configure the SVLAN interface set in the dynamic profile.

   The interface set is created dynamically when the subscriber logs in.

   ```
   [edit dynamic-profiles profile-dhcp-ipdemux interfaces]
   user@host# edit interface-set $junos–svlan-interface-set-name
   ```

4. Include dynamic IP demux interface creation within the dynamic interface set.

   ```
   [edit dynamic-profiles profile-dhcp-ipdemux interfaces interface-set
     $junos-svlan-interface-set-name]
   user@host# set interface $junos-interface-ifd-name unit
     $junos-underlying-interface-unit
   ```

5. Access the SVLAN interface set name that you expect **$junos-svlan-interface-set-name** to generate. For example, to specify the expected interface set name for aggregated Ethernet interface ae0 and outer VLAN tag 111, include **ae0-111** for the ***interface-set-name*** variable.

   ```
   [edit class-of-service interfaces]
   user@host# edit interface-set ae0-111
   ```

6. Apply traffic shaping and queuing parameters to the SVLAN interface set.

   TIP: You must configure the interface set in the static **[edit class-of-service]** hierarchy, not in the **[edit dynamic-profiles]** hierarchy.

   ```
   [edit class-of-service interfaces interface-set ae0-111]
   user@host# set output-traffic-control-profile tcp1
   ```

7. Apply traffic shaping and queuing parameters to any remaining traffic on the SVLAN interface set.

   ```
   [edit class-of-service interfaces interface-set ae0-111]
   user@host# set output-traffic-control-profile-remaining tcp3
   ```

Results

From configuration mode, confirm your configuration by entering the **show dynamic-profiles** command and the **show class-of-service** command. If the output does not display the intended configuration, repeat the instructions in this example to correct the configuration.

```
user@host# show dynamic-profiles
dynamic-profiles {
  profile-dhcp-ipdemux {
    interfaces {
      interface-set "$junos-svlan-interface-set-name" {
        interface "$junos-interface-ifd-name" {
          unit "$junos-underlying-interface-unit";
        }
      }
      "$junos-interface-ifd-name" {
        unit "$junos-underlying-interface-unit";
      }
    }
  }
}
```

```
user@host# show class-of-service
class-of-service {
  traffic-control-profiles {
    tcp1 {
      scheduler-map schedMap;
      shaping-rate 50m;
      guaranteed-rate 200k;
    }
    tcp3 {
      inactive: scheduler-map ss1q0q1;
      shaping-rate 20m;
      guaranteed-rate 5m;
    }
  }
  interfaces {
    interface-set ae0-111 {
      output-traffic-control-profile tcp1;
      output-traffic-control-profile-remaining tcp3;
    }
  }
}
```

Verification

To confirm that the configuration is correct, perform these tasks:

Verifying the Interfaces that are Included in the Interface Set

Purpose Verify the interfaces that are included in the interface set.

Action user@host> show class-of-service interface-set

Displaying Information for Active Subscribers

Purpose Display information for active subscribers.

Action

```
user@host> show subscribers detail
```

Related Documentation

- *Dynamic Profiles Overview*
- *Configuring a Basic Dynamic Profile*
- *Configuring Hierarchical Schedulers for CoS*
- *Configuring Remaining Common Queues on MIC and MPC Interfaces*

CHAPTER 12

Configuration Statements

[edit dynamic-profiles] Hierarchy Level

```
dynamic-profiles {
  profile-name {
    class-of-service {
      interfaces {
        interface-name {
          unit logical-unit-number {
            classifiers {
              type (classifier-name | default);
            }
            output-traffic-control-profile (profile-name |
              $junos-cos-traffic-control-profile);
            rewrite-rules {
              dscp (rewrite-name | default);
              dscp-ipv6 (rewrite-name | default);
              ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
              inet-precedence (rewrite-name | default);
              }
            }
          }
        }
      }
      scheduler-maps {
        map-name {
          forwarding-class class-name scheduler scheduler-name;
        }
      }
      schedulers {
        (scheduler-name) {
          buffer-size (percent percentage | remainder | temporal microseconds |
            $junos-cos-scheduler-bs);
          drop-profile-map loss-priority (any | low | medium-low | medium-high | high)
            protocol (any | non-tcp | tcp) drop-profile (profile-name | predefined-variable);
          excess-priority (low | high | $junos-cos-scheduler-excess-priority);
          excess-rate (percent percentage  | percent $junos-cos-scheduler-excess-rate);
          overhead-accounting (shaping-mode) <bytes (byte-value>;
          priority (priority-level | $junos-cos-scheduler-priority);
          shaping-rate (rate | predefined-variable);
```

```
        transmit-rate (rate | percent percentage | remainder | percent percentage
          $junos-cos-scheduler-tx) <exact | rate-limit>;
      }
    }
    traffic-control-profiles profile-name {
      delay-buffer-rate (percent percentage | rate);
      excess-rate (percent percentage | proportion value | percent
        $junos-cos-excess-rate);
      guaranteed-rate (percent percentage | rate);
      overhead-accounting (shaping-mode) <bytes (byte-value>;
      scheduler-map map-name;
      shaping-rate (percent percentage | rate | predefined-variable);
    }
  }
  firewall {
    family family {
      fast-update-filter filter-name {
        interface-specific;
      match-order [match-order];
        term term-name {
          from {
            match-conditions;
          }
          then {
            action;
            action-modifiers;
          }
        only-at-create;
      filter filter-name {
      interface-specific;
        term term-name {
          from {
            match-conditions;
          }
          then {
            action;
            action-modifiers;
          }
        }
    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 {
        policer-action;
      }
    }
    hierarchical-policer policer-name {
      aggregate {
        if-exceeding {
          bandwidth-limit-limit bps;
```

```
            burst-size-limit bytes;
          }
          then {
            policer-action;
          }
        }
        premium {
          if-exceeding {
            bandwidth-limit bps;
            burst-size-limit bytes;
          }
          then {
            policer-action;
          }
        }
      }
      three-color-policer policer-name {
        action {
          loss-priority high then discard;
        }
        logical-interface-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;
          }
        }
      }
}
policy-options {
  prefix-listname {
    ip-addresses;
  }
}
interfaces {
  interface-name {
    unit logical-unit-number {
      family family {
        access-concentrator name;
        address address;
        direct-connect;
        duplicate-protection;
        dynamic-profile profile-name;
        filter {
          adf {
            counter;
            input-precedence precedence;
            not-mandatory;
```

```
            output-precedence precedence;
            rule rule-value;
          }
          input filter-name {
            precedence precedence;
            shared-name filter-shared-name;
          }
          output filter-name {
            precedence precedence;
            shared-name filter-shared-name;
          }
        }
        max-sessions number;
        max-sessions-vsa-ignore;
        rpf-check {
          fail-filter filter-name;
          mode loose;
        }
        service {
          input {
            service-set service-set-name {
              service-filter filter-name;
            }
            post-service-filter filter-name;
          }
          output {
            service-set service-set-name {
              service-filter filter-name;
            }
          }
        }
        service-name-table table-name;
        short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
          maximum-seconds>;
        unnumbered-address interface-name <preferred-source-address address>;
      }
      ppp-options {
        chap;
        pap;
      }
      vlan-id number;
    }
    vlan-tagging;
  }
  interface-set interface-set-name {
    interface interface-name {
      unit logical-unit-number;
    }
  }
  demux0 {
    unit logical-unit-number {
      demux-options {
        underlying-interface interface-name
      }
      demux-source {
        source-prefix;
```

```
        }
        family family {
          access-concentrator name;
          address address;
          direct-connect;
          duplicate-protection;
          dynamic-profile profile-name;
          filter {
            input filter-name;
            output filter-name;
          }
          mac-validate (loose | strict):
          max-sessions number;
          max-sessions-vsa-ignore;
          service-name-table table-name;
          short-cycle-protection <lockout-time-min minimum-seconds lockout-time-max
            maximum-seconds>;
          unnumbered-address interface-name <preferred-source-address address>;
        }
      }
    }
    pp0 {
      unit logical-unit-number {
        keepalives interval seconds;
        no-keepalives;
        pppoe-options {
          underlying-interface interface-name;
          server;
        }
        ppp-options {
          authentication [ authentication-protocols ];
          chap {
            challenge-length minimum minimum-length maximum maximum-length;
          }
          pap;
        }
        family inet {
          unnumbered-address interface-name address;
          address address;
          service {
            input {
              service-set service-set-name {
                service-filter filter-name;
              }
              post-service-filter filter-name;
            }
            output {
              service-set service-set-name {
                service-filter filter-name;
              }
            }
          }
          filter {
            input filter-name {
              precedence precedence;
            }
```

```
                output filter-name {
                    precedence precedence;
                }
            }
        }
    }
}
protocols {
    igmp {
        interface interface-name {
            accounting;
            disable;
            group-policy;
            immediate-leave
            no-accounting;
            promiscuous-mode;
            ssm-map ssm-map-name;
            static {
                group group {
                    source source;
                }
            }
            version version;
        }
        mld {
            interface interface-name {
                disable;
                (accounting | no-accounting);
                group-policy;
                immediate-leave;
                oif-map;
                passive;
                ssm-map ssm-map-name;
                static {
                    group multicast-group-address {
                        exclude;
                        group-count number;
                        group-increment increment;
                        source ip-address {
                            source-count number;
                            source-increment increment;
                        }
                    }
                }
                version version;
            }
        }
        router-advertisement {
            interface interface-name {
                current-hop-limit number;
                default-lifetime seconds;
                (managed-configuration | no-managed-configuration);
                max-advertisement-interval seconds;
                min-advertisement-interval seconds;
                (other-stateful-configuration | no-other-stateful-configuration);
```

```
            prefix prefix {
              (autonomous | no-autonomous);
              (on-link | no-on-link);
              preferred-lifetime seconds;
              valid-lifetime seconds;
            }
            reachable-time milliseconds;
            retransmit-timer milliseconds;
          }
        }
      }
    }
  }
  routing-instances routing-instance-name {
    interface interface-name;
    routing-options {
      access {
        route prefix {
          next-hop next-hop;
          metric route-cost;
          preference route-distance;
          tag route-tag;
        }
      }
      access-internal {
        route subscriber-ip-address {
          qualified-next-hop underlying-interface {
            mac-address address;
          }
        }
      }
      multicast {
        interface interface-name {
          no-qos-adjust;
        }
      }
    }
    rib routing-table-name {
      access {
        route prefix {
          next-hop next-hop;
          metric route-cost;
          preference route-distance;
          tag route-tag;
        }
      }
      access-internal {
        route subscriber-ip-address {
          qualified-next-hop underlying-interface {
            mac-address address;
          }
        }
      }
    }
  }
  routing-options {
```

```
    access {
      route prefix {
        next-hop next-hop;
        metric route-cost;
        preference route-distance;
        tag route-tag;
      }
    }
    access-internal {
      route subscriber-ip-address {
        qualified-next-hop underlying-interface {
          mac-address address;
        }
      }
    }
    multicast {
      interface interface-name {
        no-qos-adjust;
      }
    }
  }
  variables {
    variable-name {
      default-value default-value;
      equals expression;
      mandatory;
      uid;
      uid-reference;
    }
  }
}
```

Related Documentation

- *Dynamic Profiles Overview*
- CoS for Subscriber Access Overview on page 3
- *Configuring a Basic Dynamic Profile*
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- *Two-Color Policer Configuration Overview*
- *Three-Color Policer Configuration Overview*
- *Hierarchical Policer Configuration Overview*
- *Guidelines for Applying Traffic Policers*

buffer-size (Dynamic Scheduling)

Syntax buffer-size (percent (*percentage* | $junos-cos-scheduler-bs) | remainder | temporal (*microseconds* | $junos-cos-scheduler-bs));

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

Release Information Statement introduced in Junos OS Release 9.3.
The **$junos-cos-scheduler-bs** predefined variable introduced in Junos OS Release 9.4.

Description Specify buffer size.

Default If you do not include this statement, the default scheduler transmission rate and buffer size percentages for queues 0 through 7 are 95, 0, 0, 5, 0, 0, 0, and 0 percent.

Options

percent *percentage*—Buffer size as a percentage of total buffer.

remainder—Remaining buffer available.

temporal *microseconds*—Buffer size as a temporal value. The queuing algorithm starts dropping packets when it queues more than a computed number of bytes. This maximum is computed by multiplying the logical interface speed by the configured temporal value.

Range: The ranges vary by platform as follows:

- For IQ PICs on M320 routers: 1 through 50,000 microseconds.
- For IQ PICs on other M Series routers: 1 through 100,000 microseconds.
- For other M Series routers: 1 through 200,000 microseconds.

$junos-scheduler-bs—Junos predefined variable that is replaced with the buffer size obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- scheduler (Dynamic Scheduler Maps) on page 182

class-of-service (Dynamic Profiles)

Syntax class-of-service { ... }

Hierarchy Level [edit dynamic-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure Junos OS CoS features in a dynamic profile.

Default If you do not configure any CoS features, all packets are transmitted from output transmission queue 0.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Static Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 49
- Configuring Dynamic Hierarchical Scheduling and Queuing in a Dynamic Profile for Subscriber Access on page 51

classifiers (Dynamic CoS Application)

Syntax

```
classifiers {
  dscp (classifier-name | default);
  dscp-ipv6 (classifier-name | default);
  ieee-802.1 (classifier-name | default) vlan-tag (inner | outer)
  inet-precedence (classifier-name | default);
}
```

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

Release Information Statement introduced in Junos OS Release 10.1.

Description Apply a CoS behavior aggregate classifier to a dynamic interface. You can apply a default classifier or one that is previously defined.

Options The remaining statements are explained separately.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *classifiers (Definition)*

delay-buffer-rate (Dynamic Traffic Shaping)

Syntax delay-buffer-rate (percent *percentage* | *rate* | $junos-cos-delay-buffer-rate);

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 9.2.
The **$junos-cos-delay-buffer-rate** variable introduced in Junos OS Release 9.4.

Description Base the delay-buffer calculation on a delay-buffer rate.

Default If you do not include this statement, the delay-buffer calculation is based on the guaranteed rate if one is configured, or the shaping rate if no guaranteed rate is configured.

Options *rate*—Delay-buffer rate, in bits per second (bps). You can specify a value in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

Range: 1000 through 160,000,000,000 bps

$junos-cos-delay-buffer-rate—Junos predefined variable that is replaced with the delay-buffer rate obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Traffic Scheduling and Shaping for Subscriber Access on page 57
- output-traffic-control-profile on page 177

drop-profile (Dynamic Schedulers)

Syntax drop-profile (*profile-name* | *predefined-variable*);

Hierarchy Level [edit dynamic-profiles *profile-name* 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 in Junos OS Release 9.3.
The **$junos-cos-scheduler-dropfile-low**, **$junos-cos-scheduler-dropfile-medium-low**, **$junos-cos-scheduler-dropfile-medium-high**, **$junos-cos-scheduler-dropfile-high**, and **$junos-cos-scheduler-dropfile-any** predefined variable introduced in Junos OS Release 9.4.

Description Within the drop-profile map, specify the name of the drop profile to use for random early detection (RED) for a specific packet-loss priority (PLP) level and protocol type. A drop profile maps a fill level (fullness of a queue) to a drop probability (probability that a packet will be dropped). 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.

You enable RED by applying a drop profile to a scheduler.

You configure drop profiles statically (at the **[edit class-of-service drop-profiles]** hierarchy level).

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

predefined-variable—One of the following Junos predefined variable that is replaced with a value obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached:

- **$junos-cos-scheduler-dropfile-low**—Name of the drop profile for PLP level **low** and protocol **any**, specified for a scheduler configured in a dynamic profile for subscriber access.
- **$junos-cos-scheduler-dropfile-medium-low**—Name of the drop profile for PLP level **medium-low** and protocol **any**, specified for a scheduler configured in a dynamic profile for subscriber access.
- **$junos-cos-scheduler-dropfile-medium-high**—Name of the drop profile for PLP level **medium-high** and protocol **any**, specified for a scheduler configured in a dynamic profile for subscriber access.
- **$junos-cos-scheduler-dropfile-high**—Name of the drop profile for PLP level **high** and protocol **any**, specified for a scheduler configured in a dynamic profile for subscriber access.
- **$junos-cos-scheduler-dropfile-lny**—Name of the drop profile for PLP level **any** and protocol **any**, specified for a scheduler configured in a dynamic profile for subscriber access.

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

Related Documentation
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- scheduler (Dynamic Scheduler Maps) on page 182
- *Configuring Drop Profile Maps for Schedulers*

drop-profile-map (Dynamic Schedulers)

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

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

Release Information
Statement introduced in Junos OS Release 9.3.

Description
Define loss priority value for 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
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- scheduler (Dynamic Scheduler Maps) on page 182

dscp (Dynamic Classifiers)

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

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

Release Information Statement introduced in Junos OS Release 10.1.

Description For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) classifier to a subscriber interface in a dynamic profile.

Options ***classifier-name***—Name of a **classifier** mapping configured at the **[edit class-of-service classifier 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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *classifiers (Definition)*

dscp (Dynamic Rewrite Rules)

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

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

Release Information Statement introduced in Junos OS Release 10.1.

Description For IPv4 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) rewrite rule to a subscriber interface in a dynamic profile.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile on page 83
- *rewrite-rules*

dscp-ipv6 (Dynamic Classifiers)

Syntax dscp-ipv6 (*classifier-name* | default);

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

Release Information Statement introduced before Junos OS Release 10.1.

Description For IPv6 traffic, apply a Differentiated Services (DiffServ) code point (DSCP) classifier to a subscriber interface in a dynamic profile.

Options ***classifier-name***—Name of a **classifier** mapping configured at the **[edit class-of-service classifier 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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *classifiers (Definition)*

dscp-ipv6 (Dynamic Rewrite Rules)

Syntax dscp-ipv6 (*rewrite-name* | default);

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

Release Information Statement introduced before Junos OS Release10.1.

Description For IPv6 traffic, apply a DSCP rewrite rule to a subscriber interface in a dynamic profile.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- *rewrite-rules*

forwarding-class (Dynamic Scheduler Maps)

Syntax forwarding-class *class-name*;

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service scheduler-maps *map-name*]

Release Information Statement introduced in Junos OS Release 9.3.

Description Associate a scheduler with a scheduler map.

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

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59

guaranteed-rate (Dynamic Traffic Shaping)

Syntax guaranteed-rate (*rate* | $junos-cos-guaranteed-rate) <burst-size [*bytes* | $junos-cos-guaranteed-rate-burst]>;

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 9.2.
The **$junos-cos-guaranteed-rate** variable introduced in Junos OS Release 9.4.
Option **burst-size** introduced in Junos OS Release 11.4.

Description Configure a guaranteed minimum rate for a logical interface.

Default If you do not include this statement and you do not include the **delay-buffer-rate** statement, the logical interface receives a minimal delay-buffer rate and minimal bandwidth equal to 2 MTU-sized packets.

Options

rate—Guaranteed rate in bits per second (bps). You can specify a value in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

Range: 1000 through 160,000,000,000 bps

$junos-cos-guaranteed-rate—Junos predefined variable that is replaced with the guaranteed rate obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

burst-size *bytes*—(Optional) Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

$junos-cos-guaranteed-rate-burst—(Optional) Variable for the burst-size that is specified for the guaranteed rate. Use this variable at the **[edit dynamic-profiles** *profile-name* **class-of-service traffic-control-profile**] hierarchy level.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Traffic Scheduling and Shaping for Subscriber Access on page 57
- output-traffic-control-profile on page 177

hierarchical-scheduler (Subscriber Interfaces on MX Series Routers)

Syntax

```
hierarchical-scheduler {
    implicit-hierarchy;
    maximum–hierarchy–levels number;
}
```

Hierarchy Level [edit interfaces *interface-name*]

Release Information Statement introduced in Junos OS Release 10.1.
Option **implicit-hierarchy** introduced in Junos OS Release 13.1.
Support on GRE tunnel interfaces configured on physical interfaces on MICs or MPCs in MX Series routers added in Junos OS Release 13.3.

Description Configure hierarchical scheduling options on the interface.

The statement is supported on the following interfaces:

- MIC and MPC interfaces in MX Series routers
- GRE tunnel interfaces configured on physical interfaces hosted on MIC or MPC line cards in MX Series routers

To enable hierarchical scheduling on MX Series routers, configure the **hierarchical-scheduler** statement at each member physical interface level of a particular aggregated Ethernet interface as well as at that aggregated Ethernet interface level. On other routing platforms, it is enough if you include this statement at the aggregated Ethernet interface level.

Options

implicit-hierarchy—Configure three-level hierarchical scheduling. When you include the **implicit-hierarchy** option, a hierarchical relationship is formed between the CoS scheduler nodes at level 1, level 2, and level 3. The **implicit-hierarchy** option is supported only on MPC/MIC subscriber interfaces and interface sets running over aggregated Ethernet on MX Series routers.

maximum-hierarchy-levels ***number***—Configure two-level hierarchical scheduling. Specify the maximum number of hierarchical scheduling levels allowed for node scaling. The only supported value is 2. The **maximum-hierarchy-levels** option is supported on MPC/MIC or EQ DPC subscriber interfaces and interface sets running over aggregated Ethernet on MX Series routers.

- If you include the **maximum-hierarchy-levels** option, interface sets are allowed only at level 3; they are not allowed at level 2. In this case, if you configure a level 2 interface set, you generate Packet Forwarding Engine errors.
- If you do not include the **maximum-hierarchy-levels** option, interface sets can be at either level 2 or level 3, depending on whether the member logical interfaces within the interface set have a traffic control profile. If any member logical interface has a traffic control profile, then the interface set is a level 2 CoS scheduler node. If no member logical interface has a traffic control profile, the interface set is at level 3.

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

Related Documentation
- Understanding Two-Level and Three-level Hierarchical CoS for Subscriber Interfaces on MX Series Routers on page 4
- Configuring Hierarchical CoS for a Subscriber Interface of Aggregated Ethernet Links on page 86
- *Configuring Hierarchical Schedulers for CoS*
- Configuring Hierarchical CoS on a Static PPPoE Subscriber Interface on page 87
- Hierarchical CoS on MPLS Pseudowire Subscriber Interfaces Overview on page 31

ieee-802.1 (Dynamic Classifiers)

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

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

Release Information
Statement introduced in Junos OS Release 10.1.

Description
Apply an IEEE-802.1 classifier to a subscriber interface in a dynamic profile.

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

default—The default mapping.

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
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *classifiers (Definition)*

ieee-802.1 (Dynamic Rewrite Rules)

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

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

Release Information Statement introduced in Junos OS Release 10.1.

Description Apply an IEEE-802.1 rewrite rule to a subscriber interface in a dynamic profile.

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.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile on page 83
- *rewrite-rules*

inet-precedence (Dynamic Classifiers)

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

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

Release Information Statement introduced in Junos OS Release 10.1.

Description Apply a IPv4 precedence classifier to a subscriber interface in a dynamic profile.

Options

classifier-name—Name of a **classifier** mapping configured at the **[edit class-of-service classifier 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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *classifiers (Definition)*

inet-precedence (Dynamic Rewrite Rules)

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

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

Release Information Statement introduced in Junos OS Release 10.1.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile on page 83
- *rewrite-rules*

interface (Dynamic Interface Sets)

Syntax

```
interface interface-name {
  unit logical unit number {
    advisory-options {
      downstream-rate rate;
      upstream-rate rate;
    }
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces interface-set *interface-set-name*]

Release Information Statement introduced in Junos OS Release 12.2.

Description Add a subscriber interface to a dynamic interface set.

In a dynamic profile that defines an agent circuit identifier (ACI) interface set, observe the following guidelines when you use the **interface** statement:

- Use the predefined dynamic interface variable **$junos-interface-ifd-name** to represent the interface name. Do not use a specific interface name, such as **demux0**, when defining an ACI interface set.
- Do not include the **unit *logical-unit-number*** statement.

Options *interface-name*—Either the specific name of the interface to include in the interface set, or the predefined dynamic interface variable **$junos-interface-ifd-name**. The interface variable is dynamically replaced with the interface that the DHCP or PPPoE subscriber accesses when connecting to the router.

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

- *Defining Agent Circuit Identifier Interface Sets*
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring an Interface Set of Subscribers in a Dynamic Profile on page 85
- *Agent Circuit Identifier-Based Dynamic VLANs Components Overview*

interface (Dynamic Routing Options)

Syntax

```
interface interface-names {
  no-qos-adjust;
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* routing-options multicast],
[edit dynamic-profiles *profile-name* routing-instances *routing-instance-name* routing-options multicast]

Release Information

Statement introduced in Junos OS Release 9.6.

Description

Define the maximum bandwidth for a dynamic interface on which you want to apply bandwidth management.

Options

interface-name—Names of the physical or logical interface. For details about specifying interfaces, see *Types of Interfaces Overview*.

The remaining statements are explained separately.

Required Privilege Level

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

Related Documentation

- *Configuring Dynamic Access Routes for Subscriber Management*
- *Configuring Dynamic Access-Internal Routes for DHCP Subscriber Management*

interface-set (Dynamic CoS)

Syntax

```
interface-set interface-set-name {
  interface interface-name {
    unit logical-unit-number;
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name* interfaces]

Release Information Statement introduced in Junos OS Release 10.4.

Description For MX Series routers with enhanced queuing DPCs or MPC/MIC modules, configure an interface set for dynamic CoS.

Options **interface-set *interface-set-name***—Name of the scheduler to be configured or one of the following Junos OS predefined variables:

- $junos-interface-set-name—Predefined variable that, when used, is replaced with the interface-set obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.
- $junos-svlan-interface-set-name—Locally generated interface set name for use by dual-tagged VLAN interfaces based on the outer tag of the dual-tagged VLAN. The format of the generated variable is ***physical_interface_name - outer_VLAN_tag***.
- $junos-tagged-vlan-interface-set-name—Locally generated interface set name used for grouping logical interfaces stacked over logical stacked VLAN demux interfaces for either a 1:1 (dual-tagged; individual client) VLAN or N:1 (single tagged; service) VLAN. The format of the generated variable differs with VLAN type. For dual-tagged (client) VLANs, the format of the generated variable is ***physical_interface_name - outer_VLAN_tag - inner_VLAN_tag***. For single tagged (service) VLAN, the format of the generated variable is ***physical_interface_name - VLAN_tag***.

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

- CoS for Interface Sets of Subscribers Overview on page 17
- Configuring an Interface Set of Subscribers in a Dynamic Profile on page 85
- Example: Configuring a Dynamic Service VLAN Interface Set of Subscribers in a Dynamic Profile on page 143

interfaces (Dynamic CoS Definition)

Syntax

```
interfaces {
  interface-name {
    unit logical-unit-number {
      classifiers {
        dscp (classifier-name | default);
        dscp-ipv6 (classifier-name | default);
        ieee-802.1 (classifier-name | default) vlan-tag (inner | outer)
        inet-precedence (classifier-name | default);
      }
      output-traffic-control-profile (profile-name | $junos-cos-traffic-control-profile);
      rewrite-rules {
        dscp (rewrite-name | default);
        dscp-ipv6 (rewrite-name | default);
        ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
        inet-precedence (rewrite-name | default);
      }
    }
  }
}
```

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

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure interface-specific CoS properties for incoming packets.

Options *interface-name*—Either the specific name of the interface you want to assign to the dynamic profile or the interface variable ($junos-interface-ifd-name). The interface variable is dynamically replaced with the interface the client accesses when connecting to the router.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile on page 81

loss-priority (Dynamic Schedulers)

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

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

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify a loss priority to which to apply a drop profile in a dynamic 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.

high—The drop profile applies to packets with high 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.

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

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59

output-traffic-control-profile (Dynamic CoS Definition)

Syntax output-traffic-control-profile (*profile-name* | $junos-cos-traffic-control-profile);

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

Release Information Statement introduced in Junos OS Release 9.2.
Variable **$junos-cos-traffic-control-profile** introduced in Junos OS Release 11.2.

Description Apply an output traffic scheduling and shaping profile to the logical interface.

Options ***profile-name***—Name of the traffic-control profile to be applied to this interface

$junos-cos-traffic-control-profile—Variable for the traffic-control profile that is specified for the logical interface. The variable is replaced with the traffic-control profile when the subscriber is authenticated at login.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile on page 81
- *Using the CLI to Modify Traffic-Control Profiles That Are Currently Applied to Subscribers*
- traffic-control-profiles on page 187

priority (Dynamic Schedulers)

Syntax priority (*priority-level* | $junos-cos-scheduler-priority);

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

Release Information Statement introduced in Junos OS Release 9.3.
The **$junos-cos-scheduler-priority** predefined variable introduced in Junos OS Release 9.4.

Description Specify packet-scheduling priority value in a dynamic profile.

Options ***priority-level***—one of the following packet-scheduling priority values:

- **low**—Scheduler has low priority.
- **medium-low**—Scheduler has medium-low priority.
- **medium-high**—Scheduler has medium-high priority.
- **high**—Scheduler has high priority. Assigning high priority to a queue prevents the queue from being underserved.
- **strict-high**—Scheduler has strictly high priority. Configure a **high** priority queue with unlimited transmission bandwidth available to it. As long as it has traffic to send, the **strict-high** priority queue receives precedence over **low**, **medium-low**, and **medium-high** priority queues, but not **high** priority queues. You can configure **strict-high** priority on only one queue per interface.

$junos-cos-scheduler-priority—Junos predefined variable that is replaced with the packet-scheduling priority value obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- *Dynamic Variables Overview*
- scheduler (Dynamic Scheduler Maps) on page 182

protocol (Dynamic Schedulers)

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

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

Release Information Statement introduced in Junos OS Release 9.3.

Description Specify the protocol type for the specified scheduler in a dynamic profile.

Options

any—Accept any protocol type.

non-tcp—Accept any protocol type other than TCP/IP.

tcp—Accept only TCP/IP protocol.

NOTE: Protocol types non-tcp and tcp are not supported on MX Series routers.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59

rewrite-rules (Dynamic CoS Interfaces)

Syntax

```
rewrite-rules {
  dscp (rewrite-name | default);
  dscp-ipv6 (rewrite-name | default);
  ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
  inet-precedence (rewrite-name | default);
  }
}
```

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

Release Information Statement introduced in Junos OS Release 10.1.

Description Associate a rewrite-rules configuration or default mapping with a specific interface in a dynamic profile.

Options

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

default—The 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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- *rewrite-rules*

routing-options (Dynamic Profiles)

Syntax

```
routing-options {
  access {
    route prefix {
      metric route-cost;
      next-hop next-hop;
      preference route-distance;
      tag route-tag;
    }
  }
  access-internal {
    route subscriber-ip-address {
      qualified-next-hop underlying-interface {
        mac-address address;
      }
    }
  }
  multicast {
    interface interface-name {
      no-qos-adjust;
    }
  }
  rib routing-table-name {
    access {
      route prefix {
        metric route-cost;
        next-hop next-hop;
        preference route-distance;
        tag route-tag;
      }
    }
    access-internal {
      route subscriber-ip-address {
        qualified-next-hop underlying-interface {
          mac-address address;
        }
      }
    }
  }
}
```

Hierarchy Level [edit dynamic-profiles *profile-name*],
[edit dynamic-profiles *profile-name* routing-instances $junos-routing-instance]

Release Information Statement introduced in Junos OS Release 9.6.
Support at the **[edit dynamic-profiles *profile-name* routing-instances $junos-routing-instance]** hierarchy level introduced in Junos OS Release 10.1.

Description Configure protocol-independent routing properties in a dynamic profile.

The remaining statements are explained separately.

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

Related Documentation
- *Configuring Dynamic Access Routes for Subscriber Management*
- *Configuring Dynamic Access-Internal Routes for DHCP Subscriber Management*

scheduler (Dynamic Scheduler Maps)

Syntax
scheduler *scheduler-name*;

Hierarchy Level
[edit dynamic-profiles *profile-name* class-of-service scheduler-maps *map-name* forwarding-class *class-name*]

Release Information
Statement introduced in Junos OS Release 9.2.

Description
Associate a scheduler with a scheduler map in a dynamic profile.

Options
scheduler-name—Either the specific name of the scheduler configuration block or the scheduler variable (**$junos-cos-scheduler**).

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

Related Documentation
- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- *Dynamic Variables Overview*

scheduler-map (Dynamic Traffic Shaping)

Syntax scheduler-map (*map-name*);

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service traffic-control-profiles *profile-name*]

Release Information Statement introduced in Junos OS Release 9.3.
The **$junos-cos-scheduler-map** variable introduced in Junos OS Release 9.4.

Description Associate a scheduler map name with a traffic-control profile in a dynamic profile.

The scheduler map can be defined dynamically (at the **[edit dynamic-profiles *profile-name* class-of-service scheduler-maps]** hierarchy level) or statically (at the **[edit class-of-service scheduler-maps]** hierarchy level).

Options *map-name*—Name of the scheduler map or the Junos predefined variable (**$junos-cos-scheduler-map**). When you specify the variable, the scheduler-map name is obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Traffic Scheduling and Shaping for Subscriber Access on page 57
- output-traffic-control-profile on page 177

scheduler-maps (Dynamic CoS Definition)

Syntax

```
scheduler-maps {
  map-name {
    forwarding-class class-name scheduler scheduler-name;
  }
}
```

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

Release Information Statement introduced in Junos OS Release 9.3.
Support at the [**edit dynamic-profiles *profile-name***] hierarchy level introduced in Junos OS Release 9.3.

Description Specify a scheduler map name in a dynamic profile and associate it with the scheduler configuration and forwarding class.

Options *map-name*—Name of the scheduler map.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59

schedulers (Dynamic CoS Definition)

Syntax

```
schedulers {
  scheduler-name{
    adjust-minimum rate;
    adjust-percent percentage;
    buffer-size (percent percentage | remainder | temporal microseconds |
      $junos-cos-scheduler-bs);
    drop-profile-map loss-priority (any | low | medium-low | medium-high | high) protocol
      (any | non-tcp | tcp) drop-profile (profile-name | predefined-variable);
    excess-priority (low | high | $junos-cos-scheduler-excess-priority | none);
    excess-rate (percent percentage  | percent $junos-cos-scheduler-excess-rate);
    priority (priority-level | $junos-cos-scheduler-priority);
    shaping-rate (rate | predefined-variable) <burst-size  bytes>;
    transmit-rate (rate | percent percentage | remainder | percent percentage
      $junos-cos-scheduler-tx) <exact | rate-limit>;
  }
}
```

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

Release Information Statement introduced in Junos OS Release 9.3.
The **$junos-cos-scheduler** predefined variable introduced in Junos OS Release 9.4.

Description Specify scheduler name and parameter values in a dynamic profile.

Options *scheduler-name*—Name of the scheduler to be configured or the Junos OS predefined variable (**$junos-cos-scheduler**). The predefined variable is replaced with the scheduler name obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- scheduler on page 182

shaping-rate (Dynamic Traffic Shaping and Scheduling)

Syntax shaping-rate (*rate* | *predefined-variable*) <burst-size *bytes* | $junos-cos-shaping-rate-burst>;

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service traffic-control-profiles *profile-name*], [edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]

Release Information Statement introduced in Junos OS Release 9.2.
The **$junos-cos-shaping-rate** variable for traffic-control profiles introduced in Junos OS Release 9.4.
The **$junos-cos-scheduler-shaping-rate** variable for schedulers introduced in Junos OS Release 10.2.
Option **burst-size** introduced in Junos OS Release 11.4.

Description Configure a shaping rate for a logical interface or a scheduler. The sum of the shaping rates for all logical interfaces on the physical interface can exceed the physical interface bandwidth. This practice is known as oversubscription of the peak information rate (PIR).

Options *rate*—Peak rate in bits per second (bps). You can specify the value as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

Range: 1000 through 160,000,000,000 bps

predefined-variable—One of the following Junos predefined variables. The variable is replaced with a value obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

- **$junos-cos-shaping-rate**—Variable for the shaping rate that is specified for the logical interface. Use this variable at the **[edit dynamic-profiles *profile-name* class-of-service traffic-control-profiles *profile-name*]** hierarchy level.
- **$junos-cos-scheduler-shaping-rate**—Variable for the shaping rate that is specified for a scheduler. Use this variable at the **[edit dynamic-profiles *profile-name* class-of-service schedulers *scheduler-name*]** hierarchy level.

burst-size *bytes*—(Optional) Maximum burst size, in bytes.

Range: 0 through 1,000,000,000

$junos-cos-shaping-rate-burst—(Optional) Variable for the burst-size that is specified for the shaping rate. Use this variable at the **[edit dynamic-profiles *profile-name* class-of-service traffic-control-profile]** hierarchy level.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Traffic Scheduling and Shaping for Subscriber Access on page 57
- output-traffic-control-profile on page 177

traffic-control-profiles (Dynamic CoS Definition)

Syntax

```
traffic-control-profiles profile-name {
  adjust-minimum rate;
  delay-buffer-rate (percent percentage | rate);
  excess-rate (percent percentage | proportion value | percent $junos-cos-excess-rate);
  excess-rate-high (percent percentage | proportion value);
  excess-rate-low (percent percentage | proportion value);
  guaranteed-rate (percent percentage | rate) <burst-size bytes>;
  overhead-accounting (frame-mode | cell-mode) <bytes byte-value>;
  scheduler-map map-name;
  shaping-rate (percent percentage | rate | predefined-variable) <burst-size bytes>;
}
```

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

Release Information Statement introduced in Junos OS Release 9.2.

Description Configure traffic shaping and scheduling profiles.

Options ***profile-name***—Name of the traffic-control 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

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Traffic Scheduling and Shaping for Subscriber Access on page 57
- *Using the CLI to Modify Traffic-Control Profiles That Are Currently Applied to Subscribers*
- output-traffic-control-profile on page 177

transmit-rate (Dynamic Schedulers)

Syntax transmit-rate (*rate* | percent *percentage* | remainder | percent *percentage* $junos-cos-scheduler-tx) <exact | rate-limit>;

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

Release Information Statement introduced in Junos OS Release 9.3.
The **$junos-cos-scheduler-tx** predefined variable introduced in Junos OS Release 9.4.

Description Specify the transmit rate or percentage for a scheduler in a dynamic profile.

Default If you do not include this statement, the default scheduler transmission rate and buffer size percentages for queues 0 through 7 are 95, 0, 0, 5, 0, 0, 0, and 0 percent.

Options

rate—Transmission rate, in bps. You can specify a value in bits per second either as a complete decimal number or as a decimal number followed by the abbreviation **k** (1000), **m** (1,000,000), or **g** (1,000,000,000).

Range: 3200 through 160,000,000,000 bps

percent *percentage*—Percentage of transmission capacity. A percentage of zero drops all packets in the queue.

Range: 0 through 100 percent

remainder—Use remaining rate available.

$junos-cos-scheduler-tx—Junos predefined variable that is replaced with the transmission rate obtained from the RADIUS server when a subscriber authenticates over the interface to which the dynamic profile is attached.

exact—(Optional) Enforce the exact transmission rate. Under sustained congestion, a rate-controlled queue that goes into negative credit fills up and eventually drops packets. Make sure this value never exceeds the rate-controlled amount.

rate-limit—(Optional) Limit the transmission rate to the rate-controlled amount during congestion. In contrast to the **exact** option, when there is no congestion, the scheduler with the **rate-limit** option shares unused bandwidth above the rate-controlled amount.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Configuring Schedulers in a Dynamic Profile for Subscriber Access on page 59
- scheduler on page 182

unit (Dynamic Traffic Shaping)

Syntax

```
unit logical-unit-number {
  classifiers {
    type (classifier-name | default);
  }
  output-traffic-control-profile (profile-name | $junos-cos-traffic-control-profile);
  rewrite-rules {
    dscp (rewrite-name | default);
    dscp-ipv6 (rewrite-name | default);
    ieee-802.1 (rewrite-name | default) vlan-tag (outer | outer-and-inner);
    inet-precedence (rewrite-name | default);
    }
  }
}
```

Hierarchy Level

[edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name*],
[edit dynamic-profiles *profile-name* interfaces interface-set *interface-set-name* interface *interface-name*]

Release Information

Statement introduced in Junos OS Release 9.2.
Support at the **[edit dynamic-profiles *profile-name* class-of-service interfaces interface-set *interface-set-name*]** hierarchy level introduced in Junos OS Release 10.4.

Description

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

Options

logical-unit-number—One of the following options:

- **$junos-underlying-interface-unit**—For static VLANs, the unit number variable. The static unit number variable is dynamically replaced with the client unit number when the client session begins. The client unit number is specified by the DHCP when it accesses the subscriber network.
- **$junos-interface-unit**—For dynamic demux and dynamic PPPoE interfaces, the unit number variable. The static unit number variable is dynamically replaced with the client unit number when the client session begins. The client unit number is specified by the DHCP or PPP when it accesses the subscriber network.
- *value*—Specific unit number of the interface you want to assign to the dynamic-profile

Range: 0 through 16385. For demux and PPPoE interfaces, the unit numbers can range from 0 through 1,073,741,823.

The remaining statements are explained separately. The **classifiers**, **output-traffic-control-profile**, and **rewrite-rules** statements are not supported for interface sets.

Required Privilege Level

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying Traffic Shaping and Scheduling to a Subscriber Interface in a Dynamic Profile on page 81
- Configuring an Interface Set of Subscribers in a Dynamic Profile on page 85

vlan-tag (Dynamic Classifiers)

Syntax vlan-tag (inner | outer);

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* classifiers ieee-802.1]

Release Information Statement introduced in Junos OS Release 10.1.

Description Apply this IEEE-802.1 classifier to the inner or outer VLAN tags in a dynamic profile.

Default If you do not include this statement, the classifier applies to the outer VLAN tag only.

Options **inner**—Apply the classifier to the inner VLAN tag only.

outer—Apply the classifier to the outer VLAN tag only.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Classifier to a Subscriber Interface in a Dynamic Profile on page 84
- *classifiers (Definition)*

vlan-tag (Dynamic Rewrite Rules)

Syntax vlan-tag (outer | outer-and-inner);

Hierarchy Level [edit dynamic-profiles *profile-name* class-of-service interfaces *interface-name* unit *logical-unit-number* rewrite-rules ieee-802.1]

Release Information Statement introduced in Junos OS Release 10.1.

Description Apply this IEEE-802.1 rewrite rule to the outer or outer and inner VLAN tags in a dynamic profile.

Default If you do not include this statement, the rewrite rule applies to the outer VLAN tag only.

Options **outer**—Apply the rewrite rule to the outer VLAN tag only.

outer-and-inner—Apply the rewrite rule to both the outer and inner VLAN tags.

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

Related Documentation

- Guidelines for Configuring Dynamic CoS for Subscriber Access on page 41
- Applying a Rewrite Rule Definition to a Subscriber Interface in a Dynamic Profile on page 83
- *rewrite-rules*

PART 3

Administration

CHAPTER 13

Monitoring CoS for Subscriber Access

Verifying the Scheduling and Shaping Configuration for Subscriber Access

Purpose View the class-of-service (CoS) configurations that are referenced in a dynamic profile for subscriber access.

Action

- To display the entire CoS configuration, including static and dynamic parameters:

  user@host> show class-of-service

- To display the CoS configuration for a subscriber interface:

  user@host> show class-of-service interface

- To display traffic shaping and scheduling profiles:

  user@host> show class-of-service traffic-control-profile

- To display the mapping of schedulers to forwarding classes and a summary of scheduler parameters for each entry:

  user@host> show class-of-service scheduler-map

CHAPTER 14

Monitoring Commands

show class-of-service

Syntax show class-of-service

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.

Description Display the entire class-of-service (CoS) configuration, including system-chosen defaults. Executing this command is equivalent to executing all **show class-of-service** commands in succession.

Options This command has no options.

Required Privilege Level view

List of Sample Output show class-of-service on page 198

Output Fields See the output field descriptions for the commands.

Sample Output

show class-of-service

```
user@host> show class-of-service
Forwarding class                      Queue
  best-effort                             0
  expedited-forwarding                    1
  assured-forwarding                      2
  network-control                         3
Code point type: dscp
  Alias              Bit pattern
  af11               001010
  af12               001100
  af13               001110
...
Code point type: dscp-ipv6
  Alias              Bit pattern
  af11               001010
  af12               001100
  af13               001110
...
Code point type: exp
  Alias              Bit pattern
  af11               100
  af12               101
  be                 000
...
Code point type: ieee-802.1
  Alias              Bit pattern
  af11               100
  af12               101
  be                 000
...
Classifier: dscp-default, Code point type: dscp, Index: 6
  Code point         Forwarding class                  Loss priority
  000000             best-effort                       low
```

```
  000001             best-effort                        low
  000010             best-effort                        low
....
Classifier: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 7
  Code point         Forwarding class                   Loss priority
  000000             best-effort                        low
  000001             best-effort                        low
  000010             best-effort                        low
...
Loss-priority-map: frame-relay-de-default, Code point type: frame-relay-de, Index:
 12
  Code point         Loss priority
  0                  low
  1                  high

Rewrite rule: dscp-default, Code point type: dscp, Index: 23
  Forwarding class                   Loss priority       Code point
  best-effort                        low                 000000
  best-effort                        high                000000
  expedited-forwarding               low                 101110
...
Rewrite rule: dscp-ipv6-default, Code point type: dscp-ipv6, Index: 24
  Forwarding class                   Loss priority       Code point
  best-effort                        low                 000000
  best-effort                        high                000000
...
....
Drop profile: <default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
         100                 100

Scheduler map: <default>, Index: 2

  Scheduler: <default-be>, Forwarding class: best-effort, Index: 16
    Transmit rate: 95 percent, Rate Limit: none, Buffer size: 95 percent, Priority:
 low
     Drop profiles:
       Loss priority   Protocol    Index    Name
       Low             any             1    <default-drop-profile>
       Medium low      any             1    <default-drop-profile>
       Medium high     any             1    <default-drop-profile>
       High            any             1    <default-drop-profile>
...
Physical interface: fe-0/0/0, Index: 137
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2

  Logical interface: fe-0/0/0.0, Index: 69
    Object            Name                   Type                    Index
    Adaptive-shaper   fr-shaper                                      35320
    Classifier        ipprec-compatibility   ip                         11

Physical interface: fe-0/0/1, Index: 138
Queues supported: 8, Queues in use: 4
  Scheduler map: <default>, Index: 2
...
```

show class-of-service interface

Syntax show class-of-service interface
<**comprehensive** | **detail**> <*interface-name*>

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 9.0 for EX Series switches.
Forwarding class map information added in Junos OS Release 9.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 12.1 for the PTX Series Packet Transport Routers.
Command introduced in Junos OS Release 12.2 for the ACX Series Universal Access routers.
Options **detail** and **comprehensive** introduced in Junos OS Release 11.4.

Description Display the logical and physical interface associations for the classifier, rewrite rules, and scheduler map objects.

Options **none**—Display CoS associations for all physical and logical interfaces.

comprehensive—(M Series, MX Series, and T Series routers) (Optional) Display comprehensive quality-of-service (QoS) information about all physical and logical interfaces.

detail—(M Series, MX Series, and T Series routers) (Optional) Display QoS and CoS information based on the interface.

If the **interface** *interface-name* is a physical interface, the output includes:

- Brief QoS information about the physical interface
- Brief QoS information about the logical interface
- CoS information about the physical interface
- Brief information about filters or policers of the logical interface
- Brief CoS information about the logical interface

If the **interface** *interface-name* is a logical interface, the output includes:

- Brief QoS information about the logical interface
- Information about filters or policers for the logical interface
- CoS information about the logical interface

interface-name—(Optional) Display class-of-service (CoS) associations for the specified interface.

Required Privilege Level view

List of Sample Output show class-of-service interface (Physical) on page 211

show class-of-service interface (Logical) on page 212
show class-of-service interface (Gigabit Ethernet) on page 212
show class-of-service interface (PPPoE Interface) on page 212
show class-of-service interface (T4000 Routers with Type 5 FPCs) on page 212
show class-of-service interface detail on page 213
show class-of-service interface comprehensive on page 213
show class-of-service interface (ACX Series Routers) on page 224

Output Fields Table 29 on page 201 describes the output fields for the **show class-of-service interface** command. Output fields are listed in the approximate order in which they appear.

Table 29: show class-of-service interface Output Fields

Field Name	Field Description
Physical interface	Name of a physical interface.
Index	Index of this interface or the internal index of this object.
Dedicated Queues	Status of dedicated queues configured on an interface. Supported only on Trio MPC/MIC interfaces on MX Series routers.
Queues supported	Number of queues you can configure on the interface.
Queues in use	Number of queues currently configured.
Total non-default queues created	Number of queues created in addition to the default queues. Supported only on Trio MPC/MIC interfaces on MX Series routers.
Rewrite Input IEEE Code-point	(QFX Series only) IEEE 802.1p code point (priority) rewrite value. Incoming traffic from the Fibre Channel (FC) SAN is classified into the forwarding class specified in the native FC interface (NP_Port) fixed classifier and uses the priority specified as the IEEE 802.1p rewrite value.
Shaping rate	Maximum transmission rate on the physical interface. You can configure the shaping rate on the physical interface, or on the logical interface, but not on both. Therefore, the **Shaping rate** field is displayed for either the physical interface or the logical interface.
Scheduler map	Name of the output scheduler map associated with this interface.
Scheduler map forwarding class sets	(QFX Series only) Name of the fabric forwarding class set scheduler map associated with a QFabric system Interconnect device interface.
Input shaping rate	For Gigabit Ethernet IQ2 PICs, maximum transmission rate on the input interface.
Input scheduler map	For Gigabit Ethernet IQ2 PICs, name of the input scheduler map associated with this interface.
Chassis scheduler map	Name of the scheduler map associated with the packet forwarding component queues.
Rewrite	Name and type of the rewrite rules associated with this interface.
Classifier	Name and type of classifiers associated with this interface.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Forwarding-class-map	Name of the forwarding map associated with this interface.
Congestion-notification	(QFX Series only) Congestion notification state, **enabled** or **disabled**.
Logical interface	Name of a logical interface.
Object	Category of an object: **Classifier**, **Fragmentation-map** (for LSQ interfaces only), **Scheduler-map**, **Rewrite**, or **Translation Table** (for IQE PICs only).
Name	Name of an object.
Type	Type of an object: **dscp**, **dscp-ipv6**, **exp**, **ieee-802.1**, **ip**, or **inet-precedence**.
Link-level type	Encapsulation on the physical interface.
MTU	MTU size on the physical interface.
Speed	Speed at which the interface is running.
Loopback	Whether loopback is enabled and the type of loopback.
Source filtering	Whether source filtering is enabled or disabled.
Flow control	Whether flow control is enabled or disabled.
Auto-negotiation	(Gigabit Ethernet interfaces) Whether autonegotiation is enabled or disabled.
Remote-fault	(Gigabit Ethernet interfaces) Remote fault status. • **Online**—Autonegotiation is manually configured as online. • **Offline**—Autonegotiation is manually configured as offline.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Device flags	The **Device flags** field provides information about the physical device and displays one or more of the following values: • **Down**—Device has been administratively disabled. • **Hear-Own-Xmit**—Device receives its own transmissions. • **Link-Layer-Down**—The link-layer protocol has failed to connect with the remote endpoint. • **Loopback**—Device is in physical loopback. • **Loop-Detected**—The link layer has received frames that it sent, thereby detecting a physical loopback. • **No-Carrier**—On media that support carrier recognition, no carrier is currently detected. • **No-Multicast**—Device does not support multicast traffic. • **Present**—Device is physically present and recognized. • **Promiscuous**—Device is in promiscuous mode and recognizes frames addressed to all physical addresses on the media. • **Quench**—Transmission on the device is quenched because the output buffer is overflowing. • **Recv-All-Multicasts**—Device is in multicast promiscuous mode and therefore provides no multicast filtering. • **Running**—Device is active and enabled.

Interface flags	The **Interface flags** field provides information about the physical interface and displays one or more of the following values:
	• **Admin-Test**—Interface is in test mode and some sanity checking, such as loop detection, is disabled.
	• **Disabled**—Interface is administratively disabled.
	• **Down**—A hardware failure has occurred.
	• **Hardware-Down**—Interface is nonfunctional or incorrectly connected.
	• **Link-Layer-Down**—Interface keepalives have indicated that the link is incomplete.
	• **No-Multicast**—Interface does not support multicast traffic.
	• **No-receive No-transmit**—Passive monitor mode is configured on the interface.
	• **Point-To-Point**—Interface is point-to-point.
	• **Pop all MPLS labels from packets of depth**—MPLS labels are removed as packets arrive on an interface that has the **pop-all-labels** statement configured. The depth value can be one of the following:
	  ◦ **1**—Takes effect for incoming packets with one label only.
	  ◦ **2**—Takes effect for incoming packets with two labels only.
	  ◦ **[1 2]**—Takes effect for incoming packets with either one or two labels.
	• **Promiscuous**—Interface is in promiscuous mode and recognizes frames addressed to all physical addresses.
	• **Recv-All-Multicasts**—Interface is in multicast promiscuous mode and provides no multicast filtering.
	• **SNMP-Traps**—SNMP trap notifications are enabled.
	• **Up**—Interface is enabled and operational.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Flags	The **Logical interface flags** field provides information about the logical interface and displays one or more of the following values: • **ACFC Encapsulation**—Address control field Compression (ACFC) encapsulation is enabled (negotiated successfully with a peer). • **Device-down**—Device has been administratively disabled. • **Disabled**—Interface is administratively disabled. • **Down**—A hardware failure has occurred. • **Clear-DF-Bit**—GRE tunnel or IPsec tunnel is configured to clear the Don't Fragment (DF) bit. • **Hardware-Down**—Interface protocol initialization failed to complete successfully. • **PFC**—Protocol field compression is enabled for the PPP session. • **Point-To-Point**—Interface is point-to-point. • **SNMP-Traps**—SNMP trap notifications are enabled. • **Up**—Interface is enabled and operational.
Encapsulation	Encapsulation on the logical interface.
Admin	Administrative state of the interface (**Up** or **Down**)
Link	Status of physical link (**Up** or **Down**).
Proto	Protocol configured on the interface.
Input Filter	Names of any firewall filters to be evaluated when packets are received on the interface, including any filters attached through activation of dynamic service.
Output Filter	Names of any firewall filters to be evaluated when packets are transmitted on the interface, including any filters attached through activation of dynamic service.
Link flags	Provides information about the physical link and displays one or more of the following values: • **ACFC**—Address control field compression is configured. The Point-to-Point Protocol (PPP) session negotiates the ACFC option. • **Give-Up**—Link protocol does not continue connection attempts after repeated failures. • **Loose-LCP**—PPP does not use the Link Control Protocol (LCP) to indicate whether the link protocol is operational. • **Loose-LMI**—Frame Relay does not use the Local Management Interface (LMI) to indicate whether the link protocol is operational. • **Loose-NCP**—PPP does not use the Network Control Protocol (NCP) to indicate whether the device is operational. • **Keepalives**—Link protocol keepalives are enabled. • **No-Keepalives**—Link protocol keepalives are disabled. • **PFC**—Protocol field compression is configured. The PPP session negotiates the PFC option.
Hold-times	Current interface hold-time up and hold-time down, in milliseconds.
CoS queues	Number of CoS queues configured.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Last flapped	Date, time, and how long ago the interface went from down to up. The format is **Last flapped: *year-month-day hour:minute:second:timezone* (*hour:minute:second* ago)**. For example, **Last flapped: 2002-04-26 10:52:40 PDT (04:33:20 ago).**
Statistics last cleared	Number and rate of bytes and packets received and transmitted on the physical interface. • **Input bytes**—Number of bytes received on the interface. • **Output bytes**—Number of bytes transmitted on the interface. • **Input packets**—Number of packets received on the interface. • **Output packets**—Number of packets transmitted on the interface.
IPv6 transit statistics	Number of IPv6 transit bytes and packets received and transmitted on the logical interface if IPv6 statistics tracking is enabled.
Input errors	Input errors on the interface. The labels are explained in the following list: • **Errors**—Sum of the incoming frame aborts and FCS errors. • **Drops**—Number of packets dropped by the input queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. • **Framing errors**—Number of packets received with an invalid frame checksum (FCS). • **Runts**—Number of frames received that are smaller than the runt threshold. • **Giants**—Number of frames received that are larger than the giant threshold. • **Bucket Drops**—Drops resulting from the traffic load exceeding the interface transmit or receive leaky bucket configuration. • **Policed discards**—Number of frames that the incoming packet match code discarded because they were not recognized or not of interest. Usually, this field reports protocols that Junos OS does not handle. • **L3 incompletes**—Number of incoming packets discarded because they failed Layer 3 (usually IPv4) sanity checks of the header. For example, a frame with less than 20 bytes of available IP header is discarded. Layer 3 incomplete errors can be ignored by configuring the **ignore-l3-incompletes** statement. • **L2 channel errors**—Number of times the software did not find a valid logical interface for an incoming frame. • **L2 mismatch timeouts**—Number of malformed or short packets that caused the incoming packet handler to discard the frame as unreadable. • **HS link CRC errors**—Number of errors on the high-speed links between the ASICs responsible for handling the router interfaces. • **HS link FIFO overflows**—Number of FIFO overflows on the high-speed links between the ASICs responsible for handling the router interfaces.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Output errors	Output errors on the interface. The labels are explained in the following list: • **Carrier transitions**—Number of times the interface has gone from **down** to **up**. This number does not normally increment quickly, increasing only when the cable is unplugged, the far-end system is powered down and up, or another problem occurs. If the number of carrier transitions increments quickly (perhaps once every 10 seconds), the cable, the far-end system, or the PIC is malfunctioning. • **Errors**—Sum of the outgoing frame aborts and FCS errors. • **Drops**—Number of packets dropped by the output queue of the I/O Manager ASIC. If the interface is saturated, this number increments once for every packet that is dropped by the ASIC's RED mechanism. NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), the **Drops** field does not always use the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs. • **Aged packets**—Number of packets that remained in shared packet SDRAM so long that the system automatically purged them. The value in this field should never increment. If it does, it is most likely a software bug or possibly malfunctioning hardware. • **HS link FIFO underflows**—Number of FIFO underflows on the high-speed links between the ASICs responsible for handling the router interfaces. • **MTU errors**—Number of packets whose size exceeds the MTU of the interface.
Egress queues	Total number of egress queues supported on the specified interface.
Queue counters	CoS queue number and its associated user-configured forwarding class name. • **Queued packets**—Number of queued packets. • **Transmitted packets**—Number of transmitted packets. • **Dropped packets**—Number of packets dropped by the ASIC's RED mechanism. NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), the **Dropped packets** field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.
SONET alarms **SONET defects**	(SONET) SONET media-specific alarms and defects that prevent the interface from passing packets. When a defect persists for a certain period, it is promoted to an alarm. Based on the router configuration, an alarm can ring the red or yellow alarm bell on the router or light the red or yellow alarm LED on the craft interface. See these fields for possible alarms and defects: **SONET PHY**, **SONET section**, **SONET line**, and **SONET path**.
SONET PHY	Counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET PHY** field has the following subfields: • **PLL Lock**—Phase-locked loop • **PHY Light**—Loss of optical signal

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
SONET section	Counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET section** field has the following subfields: • **BIP-B1**—Bit interleaved parity for SONET section overhead • **SEF**—Severely errored framing • **LOS**—Loss of signal • **LOF**—Loss of frame • **ES-S**—Errored seconds (section) • **SES-S**—Severely errored seconds (section) • **SEFS-S**—Severely errored framing seconds (section)
SONET line	Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET line** field has the following subfields: • **BIP-B2**—Bit interleaved parity for SONET line overhead • **REI-L**—Remote error indication (near-end line) • **RDI-L**—Remote defect indication (near-end line) • **AIS-L**—Alarm indication signal (near-end line) • **BERR-SF**—Bit error rate fault (signal failure) • **BERR-SD**—Bit error rate defect (signal degradation) • **ES-L**—Errored seconds (near-end line) • **SES-L**—Severely errored seconds (near-end line) • **UAS-L**—Unavailable seconds (near-end line) • **ES-LFE**—Errored seconds (far-end line) • **SES-LFE**—Severely errored seconds (far-end line) • **UAS-LFE**—Unavailable seconds (far-end line)

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
SONET path	Active alarms and defects, plus counts of specific SONET errors with detailed information. • **Seconds**—Number of seconds the defect has been active. • **Count**—Number of times that the defect has gone from inactive to active. • **State**—State of the error. A state other than **OK** indicates a problem. The **SONET path** field has the following subfields: • **BIP-B3**—Bit interleaved parity for SONET section overhead • **REI-P**—Remote error indication • **LOP-P**—Loss of pointer (path) • **AIS-P**—Path alarm indication signal • **RDI-P**—Path remote defect indication • **UNEQ-P**—Path unequipped • **PLM-P**—Path payload (signal) label mismatch • **ES-P**—Errored seconds (near-end STS path) • **SES-P**—Severely errored seconds (near-end STS path) • **UAS-P**—Unavailable seconds (near-end STS path) • **ES-PFE**—Errored seconds (far-end STS path) • **SES-PFE**—Severely errored seconds (far-end STS path) • **UAS-PFE**—Unavailable seconds (far-end STS path)
Received SONET overhead **Transmitted SONET overhead**	Values of the received and transmitted SONET overhead: • **C2**—Signal label. Allocated to identify the construction and content of the STS-level SPE and for PDI-P. • **F1**—Section user channel byte. This byte is set aside for the purposes of users. • **K1** and **K2**—These bytes are allocated for APS signaling for the protection of the multiplex section. • **J0**—Section trace. This byte is defined for STS-1 number 1 of an STS-*N* signal. Used to transmit a 1-byte fixed-length string or a 16-byte message so that a receiving terminal in a section can verify its continued connection to the intended transmitter. • **S1**—Synchronization status. The S1 byte is located in the first STS-1 number of an STS-*N* signal. • **Z3** and **Z4**—Allocated for future use.
Received path trace **Transmitted path trace**	SONET/SDH interfaces allow path trace bytes to be sent inband across the SONET/SDH link. Juniper Networks and other router manufacturers use these bytes to help diagnose misconfigurations and network errors by setting the transmitted path trace message so that it contains the system hostname and name of the physical interface. The received path trace value is the message received from the router at the other end of the fiber. The transmitted path trace value is the message that this router transmits.
HDLC configuration	Information about the HDLC configuration. • **Policing bucket**—Configured state of the receiving policer. • **Shaping bucket**—Configured state of the transmitting shaper. • **Giant threshold**—Giant threshold programmed into the hardware. • **Runt threshold**—Runt threshold programmed into the hardware.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Packet Forwarding Engine configuration	Information about the configuration of the Packet Forwarding Engine: • **Destination slot**—FPC slot number. • **PLP byte**—Packet Level Protocol byte.
CoS information	Information about the CoS queue for the physical interface. • **CoS transmit queue**—Queue number and its associated user-configured forwarding class name. • **Bandwidth %**—Percentage of bandwidth allocated to the queue. • **Bandwidth bps**—Bandwidth allocated to the queue (in bps). • **Buffer %**—Percentage of buffer space allocated to the queue. • **Buffer usec**—Amount of buffer space allocated to the queue, in microseconds. This value is nonzero only if the buffer size is configured in terms of time. • **Priority**—Queue priority: **low** or **high**. • **Limit**—Displayed if rate limiting is configured for the queue. Possible values are **none** and **exact**. If **exact** is configured, the queue transmits only up to the configured bandwidth, even if excess bandwidth is available. If **none** is configured, the queue transmits beyond the configured bandwidth if bandwidth is available.
Forwarding classes	Total number of forwarding classes supported on the specified interface.
Egress queues	Total number of egress queues supported on the specified interface.
Queue	Queue number.
Forwarding classes	Forwarding class name.
Queued Packets	Number of packets queued to this queue.
Queued Bytes	Number of bytes queued to this queue. The byte counts vary by PIC type.
Transmitted Packets	Number of packets transmitted by this queue. When fragmentation occurs on the egress interface, the first set of packet counters shows the postfragmentation values. The second set of packet counters (displayed under the **Packet Forwarding Engine Chassis Queues** field) shows the prefragmentation values.
Transmitted Bytes	Number of bytes transmitted by this queue. The byte counts vary by PIC type.
Tail-dropped packets	Number of packets dropped because of tail drop.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description

RED-dropped packets	Number of packets dropped because of random early detection (RED).
	• (M Series and T Series routers only) On M320 and M120 routers and the T Series routers, the total number of dropped packets is displayed. On all other M Series routers, the output classifies dropped packets into the following categories:
	  • **Low, non-TCP**—Number of low-loss priority non-TCP packets dropped because of RED.
	  • **Low, TCP**—Number of low-loss priority TCP packets dropped because of RED.
	  • **High, non-TCP**—Number of high-loss priority non-TCP packets dropped because of RED.
	  • **High, TCP**—Number of high-loss priority TCP packets dropped because of RED.
	• (MX Series routers with enhanced DPCs, and T Series routers with enhanced FPCs only) The output classifies dropped packets into the following categories:
	  • **Low**—Number of low-loss priority packets dropped because of RED.
	  • **Medium-low**—Number of medium-low loss priority packets dropped because of RED.
	  • **Medium-high**—Number of medium-high loss priority packets dropped because of RED.
	  • **High**—Number of high-loss priority packets dropped because of RED.
	NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.

RED-dropped bytes	Number of bytes dropped because of RED. The byte counts vary by PIC type.
	• (M Series and T Series routers only) On M320 and M120 routers and the T Series routers, only the total number of dropped bytes is displayed. On all other M Series routers, the output classifies dropped bytes into the following categories:
	  • **Low, non-TCP**—Number of low-loss priority non-TCP bytes dropped because of RED.
	  • **Low, TCP**—Number of low-loss priority TCP bytes dropped because of RED.
	  • **High, non-TCP**—Number of high-loss priority non-TCP bytes dropped because of RED.
	  • **High, TCP**—Number of high-loss priority TCP bytes dropped because of RED.
	NOTE: Due to accounting space limitations on certain Type 3 FPCs (which are supported in M320 and T640 routers), this field does not always display the correct value for queue 6 or queue 7 for interfaces on 10-port 1-Gigabit Ethernet PICs.

Transmit rate	Configured transmit rate of the scheduler. The rate is a percentage of the total interface bandwidth.
Rate Limit	Rate limiting configuration of the queue. Possible values are : • **None**—No rate limit. • **exact**—Queue transmits at the configured rate.
Buffer size	Delay buffer size in the queue.
Priority	Scheduling priority configured as **low** or **high**.
Excess Priority	Priority of the excess bandwidth traffic on a scheduler: **low**, **medium-low**, **medium-high**, **high**, or **none**.

Table 29: show class-of-service interface Output Fields *(continued)*

Field Name	Field Description
Drop profiles	Display the assignment of drop profiles. • **Loss priority**—Packet loss priority for drop profile assignment. • **Protocol**—Transport protocol for drop profile assignment. • **Index**—Index of the indicated object. Objects that have indexes in this output include schedulers and drop profiles. • **Name**—Name of the drop profile. • **Type**—Type of the drop profile: **discrete** or **interpolated**. • **Fill Level**—Percentage fullness of a queue. • **Drop probability**—Drop probability at this fill level.
Excess Priority	Priority of the excess bandwidth traffic on a scheduler.
Drop profiles	Display the assignment of drop profiles. • **Loss priority**—Packet loss priority for drop profile assignment. • **Protocol**—Transport protocol for drop profile assignment. • **Index**—Index of the indicated object. Objects that have indexes in this output include schedulers and drop profiles. • **Name**—Name of the drop profile. • **Type**—Type of the drop profile: **discrete** or **interpolated**. • **Fill Level**—Percentage fullness of a queue. • **Drop probability**—Drop probability at this fill level.
Adjustment information	Display the assignment of shaping-rate adjustments on a scheduler node or queue. • **Adjusting application**—Application that is performing the shaping-rate adjustment. ◦ The adjusting application can appear as **ancp LS-0**, which is the Junos OS Access Node Control Profile process (**ancpd**) that performs shaping-rate adjustments on schedule nodes. ◦ The adjusting application can also appear as **pppoe**, which adjusts the shaping-rate and overhead-accounting class-of-service attributes on dynamic subscriber interfaces in a broadband access network based on access line parameters in Point-to-Point Protocol over Ethernet (PPPoE) Tags [TR-101]. This feature is supported on MPC/MIC interfaces on MX Series routers. The shaping rate is based on the actual-data-rate-downstream attribute. The overhead accounting value is based on the access-loop-encapsulation attribute and specifies whether the access loop uses Ethernet (frame mode) or ATM (cell mode). • **Adjustment type**—Type of adjustment: **absolute** or **delta**. • **Configured shaping rate**—Shaping rate configured for the scheduler node or queue. • **Adjustment value**—Value of adjusted shaping rate. • **Adjustment target**—Level of shaping-rate adjustment performed: **node** or **queue**. • **Adjustment overhead-accounting mode**—Configured shaping mode: **frame** or **cell**.

Sample Output

show class-of-service interface (Physical)

```
user@host> show class-of-service interface so-0/2/3
Physical interface: so-0/2/3, Index: 135
Queues supported: 8, Queues in use: 4
```

```
Total non-default queues created: 4
  Scheduler map: <default>, Index: 2032638653

  Logical interface: fe-0/0/1.0, Index: 68, Dedicated Queues: no
    Shaping rate: 32000
    Object                  Name                  Type           Index
    Scheduler-map           <default>                             27
    Rewrite                 exp-default           exp             21
    Classifier              exp-default           exp             5
    Classifier              ipprec-compatibility  ip              8
    Forwarding-class-map    exp-default           exp             5
```

show class-of-service interface (Logical)

```
user@host> show class-of-service interface so-0/2/3.0
Logical interface: so-0/2/3.0, Index: 68, Dedicated Queues: no
    Shaping rate: 32000
    Object                  Name                  Type             Index
    Scheduler-map           <default>                               27
    Rewrite                 exp-default           exp               21
    Classifier              exp-default           exp               5
    Classifier              ipprec-compatibility  ip                8
    Forwarding-class-map    exp-default           exp               5
```

show class-of-service interface (Gigabit Ethernet)

```
user@host> show class-of-service interface ge-6/2/0
Physical interface: ge-6/2/0, Index: 175
Queues supported: 4, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Input scheduler map: <default>, Index: 3
  Chassis scheduler map: <default-chassis>, Index: 4
```

show class-of-service interface (PPPoE Interface)

```
user@host> show class-of-service interface pp0.1
Logical interface: pp0.1, Index: 85
    Object                  Name                  Type           Index
    Traffic-control-profile tcp-pppoe.o.pp0.1     Output         2726446535
    Classifier              ipprec-compatibility  ip             13

    Adjusting application: PPPoE
      Adjustment type: absolute
      Adjustment value: 5000000
      Adjustment overhead-accounting mode: cell
      Adjustment target: node
```

show class-of-service interface (T4000 Routers with Type 5 FPCs)

```
user@host> show class-of-service interface xe-4/0/0
Physical interface: xe-4/0/0, Index: 153
    Queues supported: 8, Queues in use: 4
      Shaping rate: 5000000000 bps
      Scheduler map: <default>, Index: 2
      Congestion-notification: Disabled

      Logical interface: xe-4/0/0.0, Index: 77
        Object                  Name                    Type
Index
        Classifier              ipprec-compatibility    ip
  13
```

show class-of-service interface detail

```
user@host> show class-of-service interface ge-0/3/0 detail

Physical interface: ge-0/3/0, Enabled, Physical link is Up
  Link-level type: Ethernet, MTU: 1518, Speed: 1000mbps, Loopback: Disabled,
Source filtering: Disabled, Flow control: Enabled, Auto-negotiation: Enabled,
Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000

  Physical interface: ge-0/3/0, Index: 138
  Queues supported: 4, Queues in use: 5
  Shaping rate: 50000 bps
  Scheduler map: interface-schedular-map, Index: 58414
  Input shaping rate: 10000 bps
878674  Input scheduler map: schedular-map, Index: 15103
  Chassis scheduler map: <default-chassis>, Index: 4
  Congestion-notification: Disabled

  Logical interface ge-0/3/0.0
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
    inet
    mpls
Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.0      up    up   inet
                           mpls
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.0      up    up   inet
                           mpls

  Logical interface: ge-0/3/0.0, Index: 68
    Object                  Name                    Type               Index
    Rewrite                 exp-default             exp (mpls-any)       33
    Classifier              exp-default             exp                  10
    Classifier              ipprec-compatibility    ip                   13


  Logical interface ge-0/3/0.1
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ]  Encapsulation: ENET2
    inet
Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.1      up    up   inet
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.1      up    up   inet

  Logical interface: ge-0/3/0.1, Index: 69
    Object                  Name                    Type            Index
    Classifier              ipprec-compatibility    ip               13
```

show class-of-service interface comprehensive

```
user@host> show class-of-service interface ge-0/3/0 comprehensive
Physical interface: ge-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 601, Generation: 141
  Link-level type: Ethernet, MTU: 1518, Speed: 1000mbps, BPDU Error: None,
MAC-REWRITE Error: None, Loopback: Disabled, Source filtering: Disabled, Flow
control: Enabled,
  Auto-negotiation: Enabled, Remote fault: Online
  Device flags   : Present Running
  Interface flags: SNMP-Traps Internal: 0x4000
```

```
  CoS queues     : 4 supported, 4 maximum usable queues
  Schedulers     : 256
  Hold-times     : Up 0 ms, Down 0 ms
  Current address: 00:14:f6:f4:b4:5d, Hardware address: 00:14:f6:f4:b4:5d
  Last flapped   : 2010-09-07 06:35:22 PDT (15:14:42 ago)
  Statistics last cleared: Never
  Traffic statistics:
   Input  bytes  :                   0                   0 bps
   Output bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Output packets:                   0                   0 pps
   IPv6 total statistics:
    Input  bytes  :                  0
    Output bytes  :                  0
    Input  packets:                  0
    Output packets:                  0
  Ingress traffic statistics at Packet Forwarding Engine:
   Input  bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
   Drop   bytes  :                   0                   0 bps
   Drop   packets:                   0                   0 pps
  Label-switched interface (LSI) traffic statistics:
   Input  bytes  :                   0                   0 bps
   Input  packets:                   0                   0 pps
  Input errors:
    Errors: 0, Drops: 0, Framing errors: 0, Runts: 0, Policed discards: 0, L3
incompletes: 0, L2 channel errors: 0, L2 mismatch timeouts: 0, FIFO errors: 0,
Resource errors: 0
  Output errors:
    Carrier transitions: 5, Errors: 0, Drops: 0, Collisions: 0, Aged packets: 0,
 FIFO errors: 0, HS link CRC errors: 0, MTU errors: 0, Resource errors: 0
  Ingress queues: 4 supported, 5 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 af3                            0                   0                   0

    1 af2                            0                   0                   0

    2 ef2                            0                   0                   0

    3 ef1                            0                   0                   0

  Egress queues: 4 supported, 5 in use
  Queue counters:       Queued packets  Transmitted packets      Dropped packets

    0 af3                            0                   0                   0

    1 af2                            0                   0                   0

    2 ef2                            0                   0                   0

    3 ef1                            0                   0                   0

  Active alarms  : None
  Active defects : None
  MAC statistics:                      Receive         Transmit
    Total octets                             0                0
    Total packets                            0                0
    Unicast packets                          0                0
    Broadcast packets                        0                0
    Multicast packets                        0                0
```

```
    CRC/Align errors                          0                0
    FIFO errors                               0                0
    MAC control frames                        0                0
    MAC pause frames                          0                0
    Oversized frames                          0
    Jabber frames                             0
    Fragment frames                           0
    VLAN tagged frames                        0
    Code violations                           0
  Filter statistics:
    Input packet count                        0
    Input packet rejects                      0
    Input DA rejects                          0
    Input SA rejects                          0
    Output packet count                                        0
    Output packet pad count                                    0
    Output packet error count                                  0
    CAM destination filters: 0, CAM source filters: 0
  Autonegotiation information:
    Negotiation status: Complete
    Link partner:
       Link mode: Full-duplex, Flow control: Symmetric/Asymmetric, Remote fault:
 OK
    Local resolution:
      Flow control: Symmetric, Remote fault: Link OK
  Packet Forwarding Engine configuration:
    Destination slot: 0
  CoS information:
    Direction : Output
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                              %            bps     %           usec
    2 ef2                    39          19500     0            120     high
none
    Direction : Input
    CoS transmit queue               Bandwidth               Buffer Priority
Limit
                              %            bps     %           usec
    0 af3                    30           3000    45              0      low
none

Physical interface: ge-0/3/0, Enabled, Physical link is Up
  Interface index: 138, SNMP ifIndex: 601
Forwarding classes: 16 supported, 5 in use
Ingress queues: 4 supported, 5 in use
Queue: 0, Forwarding classes: af3
  Queued:
    Packets              :                      0                      0 pps
    Bytes                :                      0                      0 bps
  Transmitted:
    Packets              :                      0                       0 pps
    Bytes                :                      0                      0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                      0                      0 pps
    RED-dropped bytes    :                      0                      0 bps
Queue: 1, Forwarding classes: af2
  Queued:
    Packets              :                      0                      0 pps
    Bytes                :                      0                      0 bps
  Transmitted:
    Packets              :                      0                       0 pps
```

```
    Bytes               :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 2, Forwarding classes: ef2
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 3, Forwarding classes: ef1
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets : Not Available
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Forwarding classes: 16 supported, 5 in use
Egress queues: 4 supported, 5 in use
Queue: 0, Forwarding classes: af3
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 1, Forwarding classes: af2
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps
Queue: 2, Forwarding classes: ef2
  Queued:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
  Transmitted:
    Packets             :                    0                    0 pps
    Bytes               :                    0                    0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
```

```
    RED-dropped bytes    :                    0                    0 bps
Queue: 3, Forwarding classes: ef1
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets : Not Available
    RL-dropped packets   :                    0                    0 pps
    RL-dropped bytes     :                    0                    0 bps
    RED-dropped packets  :                    0                    0 pps
    RED-dropped bytes    :                    0                    0 bps

Packet Forwarding Engine Chassis Queues:
Queues: 4 supported, 5 in use
Queue: 0, Forwarding classes: af3
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available
Queue: 1, Forwarding classes: af2
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available
Queue: 2, Forwarding classes: ef2
  Queued:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
  Transmitted:
    Packets              :                    0                    0 pps
    Bytes                :                    0                    0 bps
    Tail-dropped packets :                    0                    0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available
Queue: 3, Forwarding classes: ef1
  Queued:
    Packets              :                 108546                    0 pps
    Bytes                :                12754752                  376 bps
  Transmitted:
    Packets              :                 108546                    0 pps
    Bytes                :                12754752                  376 bps
    Tail-dropped packets :                       0                    0 pps
    RED-dropped packets  : Not Available
    RED-dropped bytes    : Not Available

Physical interface: ge-0/3/0, Index: 138
Queues supported: 4, Queues in use: 5
  Shaping rate: 50000 bps
```

```
Scheduler map: interface-schedular-map, Index: 58414

  Scheduler: ef2, Forwarding class: ef2, Index: 39155
    Transmit rate: 39 percent, Rate Limit: none, Buffer size: 120 us, Buffer
Limit: none, Priority: high
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
  Input shaping rate: 10000 bps
  Input scheduler map: schedular-map

Scheduler map: schedular-map, Index: 15103

  Scheduler: af3, Forwarding class: af3, Index: 35058
    Transmit rate: 30 percent, Rate Limit: none, Buffer size: 45 percent, Buffer
 Limit: none, Priority: low
    Excess Priority: unspecified
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any         40582    green
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any         18928    yellow
Drop profile: green, Type: discrete, Index: 40582
  Fill level    Drop probability
         50                   0
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: yellow, Type: discrete, Index: 18928
  Fill level    Drop probability
         50                   0
        100                 100
  Chassis scheduler map: < default-drop-profile>
Scheduler map: < default-drop-profile>, Index: 4

  Scheduler: < default-drop-profile>, Forwarding class: af3, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
 Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
```



```
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100

  Scheduler: < default-drop-profile>, Forwarding class: af2, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
 Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100

  Scheduler: < default-drop-profile>, Forwarding class: ef2, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
 Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
```

```
  Fill level    Drop probability
        100                 100

  Scheduler: < default-drop-profile>, Forwarding class: ef1, Index: 25
    Transmit rate: 25 percent, Rate Limit: none, Buffer size: 25 percent, Buffer
 Limit: none, Priority: low
    Excess Priority: low
    Drop profiles:
      Loss priority   Protocol    Index    Name
      Low             any             1    < default-drop-profile>
      Medium low      any             1    < default-drop-profile>
      Medium high     any             1    < default-drop-profile>
      High            any             1    < default-drop-profile>
Drop profile: , Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
Drop profile: < default-drop-profile>, Type: discrete, Index: 1
  Fill level    Drop probability
        100                 100
  Congestion-notification: Disabled
Forwarding class                      ID      Queue  Restricted queue  Fabric
priority  Policing priority
  af3                                  0       0          0             low
            normal
  af2                                  1       1          1             low
            normal
  ef2                                  2       2          2             high
            normal
  ef1                                  3       3          3             high
            normal
  af1                                  4       4          0             low
            normal

  Logical interface ge-0/3/0.0 (Index 68) (SNMP ifIndex 152) (Generation 159)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Local statistics:
     Input  bytes  :                   0
     Output bytes  :                   0
     Input  packets:                   0
     Output packets:                   0
    Transit statistics:
     Input  bytes  :                   0                    0 bps
     Output bytes  :                   0                    0 bps
     Input  packets:                   0                    0 pps
     Output packets:                   0                    0 pps
    Protocol inet, MTU: 1500, Generation: 172, Route table: 0
      Flags: Sendbcast-pkt-to-re
      Input Filters: filter-in-ge-0/3/0.0-i,
      Policer: Input: p1-ge-0/3/0.0-inet-i
    Protocol mpls, MTU: 1488, Maximum labels: 3, Generation: 173, Route table: 0
```

```
      Flags: Is-Primary
      Output Filters: exp-filter,,,,,

Logical interface ge-1/2/0.0 (Index 347) (SNMP ifIndex 638) (Generation 156)

Forwarding class ID  Queue  Restricted queue  Fabric priority  Policing priority
   SPU priority
best-effort      0   0      0                 low              normal
    low

Aggregate Forwarding-class statistics per forwarding-class
  Aggregate Forwarding-class statistics:
   Forwarding-class statistics:

Forwarding-class best-effort statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Forwarding-class expedited-forwarding statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

IPv4 protocol forwarding-class statistics:
  Forwarding-class statistics:
  Forwarding-class best-effort statistics:

    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Forwarding-class expedited-forwarding statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0
```

```
    IPv6  protocol forwarding-class statistics:
    Forwarding-class statistics:
     Forwarding-class best-effort statistics:

    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0

Forwarding-class expedited-forwarding statistics:
    Input unicast bytes:      0
    Output unicast bytes:     0
    Input unicast packets:    0
    Output unicast packets:   0

    Input multicast bytes:    0
    Output multicast bytes:   0
    Input multicast packets:  0
    Output multicast packets: 0


 Logical interface ge-0/3/0.0 (Index 68) (SNMP ifIndex 152)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.1 ]  Encapsulation: ENET2
    Input packets : 0
    Output packets: 0

Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.0      up    up   inet  filter-in-ge-0/3/0.0-i
                           mpls                       exp-filter
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.0      up    up
                           inet  p1-ge-0/3/0.0-inet-i
                           mpls

Filter: filter-in-ge-0/3/0.0-i
Counters:
Name                                                Bytes              Packets
count-filter-in-ge-0/3/0.0-i                            0                    0

Filter: exp-filter
Counters:
Name                                                Bytes              Packets
count-exp-seven-match                                   0                    0
count-exp-zero-match                                    0                    0
Policers:
Name                                               Packets
p1-ge-0/3/0.0-inet-i                                    0


  Logical interface: ge-0/3/0.0, Index: 68
    Object                 Name                   Type                    Index

    Rewrite                exp-default            exp (mpls-any)            33

Rewrite rule: exp-default, Code point type: exp, Index: 33
```

```
  Forwarding class                  Loss priority       Code point
  af3                               low                 000
  af3                               high                001
  af2                               low                 010
  af2                               high                011
  ef2                               low                 100
  ef2                               high                101
  ef1                               low                 110
  ef1                               high                111
    Object                 Name                  Type                      Index

    Classifier             exp-default           exp                         10

Classifier: exp-default, Code point type: exp, Index: 10
  Code point         Forwarding class                  Loss priority
  000                af3                               low
  001                af3                               high
  010                af2                               low
  011                af2                               high
  100                ef2                               low
  101                ef2                               high
  110                ef1                               low
  111                ef1                               high
    Object                 Name                  Type                      Index

    Classifier             ipprec-compatibility  ip                          13

Classifier: ipprec-compatibility, Code point type: inet-precedence, Index: 13
  Code point         Forwarding class                  Loss priority
  000                af3                               low
  001                af3                               high
  010                af3                               low
  011                af3                               high
  100                af3                               low
  101                af3                               high
  110                ef1                               low
  111                ef1                               high
Forwarding class                    ID      Queue  Restricted queue  Fabric
priority  Policing priority
  af3                               0       0          0             low
            normal
  af2                               1       1          1             low
            normal
  ef2                               2       2          2             high
            normal
  ef1                               3       3          3             high
            normal
  af1                               4       4          0             low
            normal

  Logical interface ge-0/3/0.1 (Index 69) (SNMP ifIndex 154) (Generation 160)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ]  Encapsulation: ENET2
    Traffic statistics:
     Input  bytes  :                  0
     Output bytes  :                  0
     Input  packets:                  0
     Output packets:                  0
    Local statistics:
     Input  bytes  :                  0
     Output bytes  :                  0
     Input  packets:                  0
```

```
     Output packets:                    0
    Transit statistics:
     Input  bytes  :                    0                    0 bps
     Output bytes  :                    0                    0 bps
     Input  packets:                    0                    0 pps
     Output packets:                    0                    0 pps
    Protocol inet, MTU: 1500, Generation: 174, Route table: 0
      Flags: Sendbcast-pkt-to-re

  Logical interface ge-0/3/0.1 (Index 69) (SNMP ifIndex 154)
    Flags: SNMP-Traps 0x4000 VLAN-Tag [ 0x8100.2 ]  Encapsulation: ENET2
    Input packets : 0
    Output packets: 0


Interface       Admin Link Proto Input Filter         Output Filter
ge-0/3/0.1      up    up   mpls
Interface       Admin Link Proto Input Policer         Output Policer
ge-0/3/0.1      up    up
                           mpls

  Logical interface: ge-0/3/0.1, Index: 69
    Object                 Name                   Type                    Index

    Classifier             ipprec-compatibility   ip                        13

Classifier: ipprec-compatibility, Code point type: inet-precedence, Index: 13
  Code point         Forwarding class                    Loss priority
  000                af3                                 low
  001                af3                                 high
  010                af3                                 low
  011                af3                                 high
  100                af3                                 low
  101                af3                                 high
  110                ef1                                 low
  111                ef1                                 high
Forwarding class                      ID      Queue  Restricted queue  Fabric
priority  Policing priority
  af3                                  0       0          0             low
            normal
  af2                                  1       1          1             low
            normal
  ef2                                  2       2          2             high
            normal
  ef1                                  3       3          3             high
            normal
  af1                                  4       4          0             low
            normal
```

show class-of-service interface (ACX Series Routers)

```
user@host-g11# show class-of-service interface
Physical interface: at-0/0/0, Index: 130
Queues supported: 4, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: at-0/0/0.0, Index: 69
```

```
  Logical interface: at-0/0/0.32767, Index: 70

Physical interface: at-0/0/1, Index: 133
Queues supported: 4, Queues in use: 4
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled

  Logical interface: at-0/0/1.0, Index: 71

  Logical interface: at-0/0/1.32767, Index: 72

Physical interface: ge-0/1/0, Index: 146
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                    Type                    Index
Rewrite                 dscp-default            dscp                       31
Classifier              d1                      dscp                    11331
Classifier              ci                      ieee8021p                 583

  Logical interface: ge-0/1/0.0, Index: 73
Object                  Name                    Type                    Index
Rewrite                 custom-exp              exp (mpls-any)          46413

  Logical interface: ge-0/1/0.1, Index: 74

  Logical interface: ge-0/1/0.32767, Index: 75

Physical interface: ge-0/1/1, Index: 147
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                    Type                    Index
Classifier              ipprec-compatibility    ip                         13

  Logical interface: ge-0/1/1.0, Index: 76

Physical interface: ge-0/1/2, Index: 148
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                    Type                    Index
Rewrite                 ri                      ieee8021p (outer)       35392
Classifier              ci                      ieee8021p                 583

Physical interface: ge-0/1/3, Index: 149
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                    Type                    Index
Classifier              ipprec-compatibility    ip                         13

  Logical interface: ge-0/1/3.0, Index: 77
Object                  Name                    Type                    Index
Rewrite                 custom-exp2             exp (mpls-any)          53581

Physical interface: ge-0/1/4, Index: 150
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                    Type                    Index
```

```
Classifier              ipprec-compatibility   ip                        13

Physical interface: ge-0/1/5, Index: 151
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              ipprec-compatibility   ip                        13

Physical interface: ge-0/1/6, Index: 152
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              ipprec-compatibility   ip                        13

Physical interface: ge-0/1/7, Index: 153
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              d1                     dscp                      11331

Physical interface: ge-0/2/0, Index: 154
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              ipprec-compatibility   ip                        13

Physical interface: ge-0/2/1, Index: 155
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              ipprec-compatibility   ip                        13

  Logical interface: ge-0/2/1.0, Index: 78

  Logical interface: ge-0/2/1.32767, Index: 79

Physical interface: xe-0/3/0, Index: 156
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              ipprec-compatibility   ip                        13

  Logical interface: xe-0/3/0.0, Index: 80

Physical interface: xe-0/3/1, Index: 157
Queues supported: 8, Queues in use: 5
  Scheduler map: <default>, Index: 2
  Congestion-notification: Disabled
Object                  Name                   Type                      Index
Classifier              ipprec-compatibility   ip                        13

  Logical interface: xe-0/3/1.0, Index: 81

[edit]
user@host-g11#
```

show class-of-service interface-set

Syntax **show class-of-service interface-set**
<interface-set-name>

Release Information Command introduced in Junos OS Release 9.4.

Description Display the configured shaping rate and the adjusted shaping rate for each logical interface set configured for hierarchical class of service (CoS).

Options **none**—Display CoS associations for all logical interface sets.

interface-set *interface-set-name*—(Optional) Display CoS associations for the specified interface set.

Required Privilege Level view

List of Sample Output show class-of-service interface-set on page 229

Output Fields Table 30 on page 228 describes the output fields for the **show class-of-service interface-set** command. Output fields are listed in the approximate order in which they appear.

Table 30: show class-of-service interface-set Output Fields

Field Name	Field Description
Interface-set	Name of a logical interface set composed of one or more logical interfaces for which hierarchical scheduling is enabled.
Index	Index number of this interface set or the internal index number of this object.
Physical interface	Name of a physical interface.
Queues supported	Number of queues you can configure on the interface.
Queues in use	Number of queues currently configured.
Output traffic control profile	Name of the output traffic-control profile attached to the logical interface set.

Table 30: show class-of-service interface-set Output Fields *(continued)*

Field Name	Field Description
Adjusting application	Name of the application that communicates shaping-rate adjustment information to the Junos OS class-of-service process (**cosd**) on the broadband services router (BSR). The BSR uses the information from this application to perform shaping-rate adjustments on the scheduler node that manages the interface set. The adjusting application appears as **ancp LS-0** which is the Junos OS Access Node Control Profile process (**ancpd**) that performs shaping-rate adjustments on schedule nodes. The nodes are logical interface sets configured to represent subscriber local loops. When the synchronization speed of the DSL line changes, **ancpd** communicates the local loop speed to **cosd** over the default logical system, **LS-0**, and then the BSR throttles the shaping rate on the scheduler node to the loop speed. The adjusting application can also appear as **PPPoE**, which adjusts the shaping-rate and overhead-accounting class-of-service attributes on dynamic subscriber interfaces in a broadband access network based on access line parameters in Point-to-Point Protocol over Ethernet (PPPoE) Tags [TR-101]. This feature is supported on MPC/MIC interfaces on MX Series routers. The shaping rate is based on the actual data rate downstream attribute. The overhead accounting value is based on the access loop encapsulation attribute and specifies whether the access loop uses Ethernet (frame mode) or ATM (cell mode).
Adjustment type	Type of shaping-rate adjustment performed by the BSR on the scheduler node. The type of adjustment appears as **Adjustment type**, meaning that the configured shaping rate is adjusted by an absolute value as opposed to by a percentage of the configured rate.
Configured shaping rate	The maximum transmission rate on the physical interface as configured by the output traffic-control profile attached to the scheduler node.
Adjustment value	Value of the shaping-rate adjustment information sent by the adjusting application to **cosd**.
Adjustment overhead-accounting mode	Configured shaping mode: **frame** or **cell**.

Sample Output

show class-of-service interface-set

```
user@host> show class-of-service interface-set example-ifset-ge-4/0/0-7
Interface-set: example-ifset-ge-4/0/0-7, Index: 8
Physical interface: ge-4/0/0, Index: 270
Queues supported: 8, Queues in use: 8
  Output traffic control profile: example-tcp-basic-rate, Index: 11395
Adjusting application: ancp LS-0
  Adjustment type: absolute
  Configured shaping rate: 50000000
  Adjustment value: 888000
  Adjustment overhead-accounting mode: cell
```

show class-of-service scheduler-hierarchy interface

Syntax show class-of-service scheduler-hierarchy interface *interface-name* <detail>

Release Information Command introduced in Junos OS Release 13.3 for MX Series Routers.

Description For MPC/MIC interfaces only, display the scheduler hierarchy.

Options **detail**—(Optional) Display scheduler hierarchies based on the interface-set.

interface-name—Display information about a specific interface.

Required Privilege Level view

Related Documentation
- *show interfaces queue*

List of Sample Output show class-of-service scheduler-hierarchy interface on page 230

Output Fields Table 31 on page 230 describes the output fields for the **show class-of-service scheduler-hierarchy interface** command. Output fields are listed in the approximate order in which they appear.

Table 31: show class-of-service scheduler-hierarchy interface Output Fields

Field Name	Field Description
interface	Type of interface
resource	Traffic resource associated with the logical interface
shaping-rate	Actual shaping rate in bits per second
guaranteed rate	Actual guaranteed rate in bits per second
guaranteed priority	Actual queue priority in the guaranteed region (high, low, or none)
excess priority	Actual queue priority in the excess region (high, low, or none)
queue weight	Actual queue weight for excess CoS weighted round-robin
excess weight	Actual interface unit per priority weights for excess weighted round-robin

Sample Output

show class-of-service scheduler-hierarchy interface

```
user@host> show class-of-service scheduler-hierarchy interface ge-1/0/0
------------------------------------------------------------------------------
Interface/                    shaping guarnteed  guaranteed/   queue   excess
resource name                    rate      rate       excess  weight   weight
```



```
                                kbits      kbits     priority          high/low
-------------------------------------------------------------------------------
ge-1/0/0                       100000
  ge-1/0/0 RTP                 100000          0                          1   1
    be                         100000       1000     Low  Low        1
    da                           9000       2000  Medium High        1
    vi                         100000       3000  Medium None      626
    vo                         100000       4000    High High      373
    gt                         100000          0    High High        1
  ifset                         75000          0                          1   1
    ifset RTP                  100000          0                          1   1
    best-effort                100000          0     Low  Low      950
    vi                         100000          0     Low  Low       50
    ge-1/0/0.50                100000      23000                          1   1
    be                         100000       1000     Low  Low        1
    da                           9000       2000  Medium High        1
    vi                         100000       3000  Medium None      626
    vo                         100000       4000    High High      373
    gt                         100000            High High        1
  ge-1/0/0.20                   50000      40000                        750 750
    be                          50000       1000     Low  Low        1
    da                           9000       2000  Medium High        1
    vi                          50000       3000  Medium None      626
    vo                          50000       4000    High High      373
    gt                          50000   Disabled    High High        1
  ge-1/0/0.32767               100000       2000                          1   1
    best-effort                100000       1900     Low  Low      950
    vi                         100000        100     Low  Low       50
```

show class-of-service scheduler-hierarchy interface-set

Syntax show class-of-service scheduler-hierarchy interface-set *interface-set-name* <detail>

Release Information Command introduced in Junos OS Release 13.3 for MX Series Routers.

Description For MPC/MIC interface sets only, display the scheduler hierarchy.

Options **detail**—(Optional) Display scheduler hierarchies based on the interface-set.

interface-set-name—Display information about a specific interface-set.

Required Privilege Level view

Related Documentation
- *show interfaces queue*

List of Sample Output show class-of-service scheduler-hierarchy interface-set on page 232

Output Fields Table 32 on page 232 describes the output fields for the **show class-of-service scheduler-hierarchy interface-set** command. Output fields are listed in the approximate order in which they appear.

Table 32: show class-of-service scheduler-hierarchy interface-set Output Fields

Field Name	Field Description
interface	Type of interface
resource	Traffic resource associated with the logical interface
shaping-rate	Actual shaping rate in bits per second
guaranteed rate	Actual guaranteed rate in bits per second
guaranteed priority	Actual queue priority in the guaranteed region (high, low, or none)
excess priority	Actual queue priority in the excess region (high, low, or none)
queue weight	Actual queue weight for excess CoS weighted round-robin
excess weight	Actual interface-set per priority weights for excess weighted round-robin

Sample Output

show class-of-service scheduler-hierarchy interface-set

```
user@host> show class-of-service scheduler-hierarchy interface-set ifset
------------------------------------------------------------------------------
Interface/                    shaping guarnteed  guaranteed/   queue   excess
resource name                    rate      rate       excess  weight   weight
```



```
                                kbits      kbits      priority           high/low
------------------------------------------------------------------------------
ge-1/0/0                       100000
  ge-1/0/0 RTP                 100000          0                          1   1
    be                         100000       1000      Low  Low        1
    da                           9000       2000   Medium High        1
    vi                         100000       3000   Medium None      626
    vo                         100000       4000     High High      373
    gt                         100000          0     High High        1
  ge-1/0/0.20                   50000      40000                      750 750
    be                          50000       1000      Low  Low        1
    da                           9000       2000   Medium High        1
    vi                          50000       3000   Medium None      626
    vo                          50000       4000     High High      373
    gt                          50000   Disabled     High High        1
```

show class-of-service scheduler-map

Syntax show class-of-service scheduler-map
<*name*>

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.

Description Display the mapping of schedulers to forwarding classes and a summary of scheduler parameters for each entry.

Options **none**—Display all scheduler maps.

name—(Optional) Display a summary of scheduler parameters for each forwarding class to which the named scheduler is assigned.

Required Privilege Level view

List of Sample Output show class-of-service scheduler-map on page 235

Output Fields Table 33 on page 234 describes the output fields for the **show class-of-service scheduler-map** command. Output fields are listed in the approximate order in which they appear.

Table 33: show class-of-service scheduler-map Output Fields

Field Name	Field Description
Scheduler map	Name of the scheduler map.
Index	Index of the indicated object. Objects having indexes in this output include scheduler maps, schedulers, and drop profiles.
Scheduler	Name of the scheduler.
Forwarding class	Classification of a packet affecting the forwarding, scheduling, and marking policies applied as the packet transits the router.
Transmit rate	Configured transmit rate of the scheduler (in bps). The rate is a percentage of the total interface bandwidth, or the keyword **remainder**, which indicates that the scheduler receives the remaining bandwidth of the interface.
Rate Limit	Rate limiting configuration of the queue. Possible values are **none**, meaning no rate limiting, and **exact**, meaning the queue only transmits at the configured rate.
Maximum buffer delay	Amount of transmit delay (in milliseconds) or the buffer size of the queue. The buffer size is shown as a percentage of the total interface buffer allocation, or by the keyword **remainder** to indicate that the buffer is sized according to what remains after other scheduler buffer allocations.
Priority	Scheduling priority: **low** or **high**.

Table 33: show class-of-service scheduler-map Output Fields *(continued)*

Field Name	Field Description
Excess priority	Priority of excess bandwidth: **low**, **medium-low**, **medium-high**, **high**, or **none**.
Adjust minimum	Minimum shaping rate for an adjusted queue, in bps.
Adjust percent	Bandwidth adjustment applied to a queue, in percent.
Drop profiles	Table displaying the assignment of drop profiles by name and index to a given loss priority and protocol pair.
Loss priority	Packet loss priority for drop profile assignment.
Protocol	Transport protocol for drop profile assignment.
Name	Name of the drop profile.

Sample Output

show class-of-service scheduler-map

```
user@host> show class-of-service scheduler-map
Scheduler map: dd-scheduler-map, Index: 84

  Scheduler: aa-scheduler, Index: 8721, Forwarding class: aa-forwarding-class
   Transmit rate: 30 percent, Rate Limit: none, Maximum buffer delay: 39 ms,
   Priority: high
   Drop profiles:
    Loss priority   Protocol    Index    Name
     Low            non-TCP     8724     aa-drop-profile
     Low            TCP         9874     bb-drop-profile
     High           non-TCP     8833     cc-drop-profile
     High           TCP         8484     dd-drop-profile

  Scheduler: bb-scheduler, Forwarding class: aa-forwarding-class
   Transmit rate: 40 percent, Rate limit: none, Maximum buffer delay: 68 ms,
   Priority: high
   Drop profiles:
    Loss priority   Protocol    Index    Name
     Low            non-TCP     8724     aa-drop-profile
     Low            TCP         9874     bb-drop-profile
     High           non-TCP     8833     cc-drop-profile
     High           TCP         8484     dd-drop-profile
```

show class-of-service traffic-control-profile

Syntax show class-of-service traffic-control-profile
<profile-name>

Release Information Command introduced before Junos OS Release 7.4.
Command introduced in Junos OS Release 11.1 for the QFX Series.
Command introduced in Junos OS Release 12.2 for ACX Series Routers.

Description For Gigabit Ethernet IQ PICs, Channelized IQ PICs, EQ DPCs, and Trio MPC/MIC interfaces only, display traffic shaping and scheduling profiles.

(ACX Series routers) For ATM IMA pseudowire interfaces, display traffic shaping and scheduling profiles.

Options **none**—Display all profiles.

profile-name—(Optional) Display information about a single profile.

Required Privilege Level view

List of Sample Output
show class-of-service traffic-control-profile on page 238
show class-of-service traffic-control-profile (MX Series routers with Clear Channel Multi-Rate CE MIC) on page 238
show class-of-service traffic-control-profile (ACX Series routers with ATM IMA pseudowire interfaces) on page 238

Output Fields Table 34 on page 236 describes the output fields for the **show class-of-service traffic-control-profile** command. Output fields are listed in the approximate order in which they appear.

Table 34: show class-of-service traffic-control-profile Output Fields

Field Name	Field Description
Traffic control profile	Name of the traffic control profile.
Index	Index number of the traffic control profile.
ATM Service	(MX Series routers with ATM Multi-Rate CE MIC) Configured category of ATM service. Possible values: • cbr—Constant bit rate. • rtvbr—Real time variable bit rate. • nrtvbr—Non real time variable bit rate. • ubr—Unspecified bit rate.
Maximum Burst Size	Configured maximum burst size, in cells.
Peak rate	Configured peak rate, in cps.

Table 34: show class-of-service traffic-control-profile Output Fields *(continued)*

Field Name	Field Description
Sustained rate	Configured sustained rate, in cps.
Shaping rate	Configured shaping rate, in bps. NOTE: (MX Series routers with ATM Multi-Rate CE MIC) Configured peak rate, in cps.
Shaping rate burst	Configured burst size for the shaping rate, in bytes. NOTE: (MX Series routers with ATM Multi-Rate CE MIC) Configured maximum burst rate, in cells.
Shaping rate priority high	Configured shaping rate for high-priority traffic, in bps.
Shaping rate priority medium	Configured shaping rate for medium-priority traffic, in bps.
Shaping rate priority low	Configured shaping rate for low-priority traffic, in bps.
Shaping rate excess high	Configured shaping rate for high-priority excess traffic, in bps.
Shaping rate excess low	Configured shaping rate for low-priority excess traffic, in bps.
Scheduler map	Name of the associated scheduler map.
Delay Buffer rate	Configured delay buffer rate, in bps.
Excess rate	Configured excess rate, in percent or proportion.
Excess rate high	Configured excess rate for high priority traffic, in percent or proportion.
Excess rate low	Configured excess rate for low priority traffic, in percent or proportion.
Guaranteed rate	Configured guaranteed rate, in bps or cps. NOTE: (MX Series routers with ATM Multi-Rate CE MIC) This value depends on the ATM service category chosen. Possible values: • **cbr**—Guaranteed rate is equal to the configured peak rate in cps. • **rtvbr**—Guaranteed rate is equal to the configured sustained rate in cps. • **nrtvbr**—Guaranteed rate is equal to the configured sustained rate in cps.
Guaranteed rate burst	Configured burst size for the guaranteed rate, in bytes.
adjust-minimum	Configured minimum shaping rate for an adjusted queue, in bps.

Table 34: show class-of-service traffic-control-profile Output Fields *(continued)*

Field Name	Field Description
overhead accounting mode	Configured shaping mode: **Frame Mode** or **Cell Mode**.
Overhead bytes	Configured byte adjustment value.

Sample Output

show class-of-service traffic-control-profile

```
user@host> show class-of-service traffic-control-profile
Traffic control profile: Profile1, Index: 57625
  Scheduler map: m1
  Delay Buffer rate: 500000
  Guaranteed rate: 1000000

Traffic control profile: Profile2, Index: 57624
  Scheduler map: m2
  Delay Buffer rate: 600000
  Guaranteed rate: 2000000

Traffic control profile: Profile3, Index: 57627
  Scheduler map: m3
  Delay Buffer rate: 800000
  Guaranteed rate: 3000000
 .Excess rate high: proportion 4

Traffic control profile: Profile4, Index: 57626
  Scheduler map: m4
  Delay Buffer rate: 750000
  Guaranteed rate: 4000000
..adjust-minimum 20000000
```

show class-of-service traffic-control-profile (MX Series routers with Clear Channel Multi-Rate CE MIC)

```
user@host> show class-of-service traffic-control-profile
Traffic control profile: at-vbr1, Index: 11395
  ATM Service: RTVBR
  Scheduler map: m3
  overhead accounting mode: Frame Mode
  Shaping rate: 1000 cps
  Shaping rate burst: 500 cells
  Delay Buffer rate: 2000 cps
  Guaranteed rate: 1000 cps

Traffic control profile: foo, Index: 38286
  ATM Service: UBR
  Scheduler map: m3
  overhead accounting mode: Frame Mode
```

show class-of-service traffic-control-profile (ACX Series routers with ATM IMA pseudowire interfaces)

```
user@host> show class-of-service traffic-control-profile
Traffic control profile: foo, Index: 38286
  ATM Service: RTVBR
  Shaping rate: 2000 cps
```

```
Shaping rate burst: 200 cells
Scheduler map: <default>
Delay Buffer rate: 1000 cps
Guaranteed rate: 1700 cps
```

PART 4

Troubleshooting

CHAPTER 15

Acquiring Troubleshooting Information

This chapter describes messages with the **COSD** prefix. They are generated by the class-of-service (CoS) process (cosd), which enables the routing platform to provide different levels of service to applications based on packet classifications.

COSD_AGGR_CONFIG_INVALID

System Log Message Error: Cannot have config *error-message interface-name*

Description The class-of-service (CoS) process (cosd) did not apply the config on this interface because it was not valid in this case.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause One possible cause is if any Class-of-Service is configured on an interface which is a part of an aggregated interface

Action Remove or change the config from/on the interface.

COSD_CHASSIS_SCHED_MAP_INVALID

System Log Message Chassis scheduler map incorrectly applied to interface *interface-name*: *error-message*

Description The class-of-service (CoS) process (cosd) did not apply a chassis scheduler map to the indicated interface, because the configuration used to apply the scheduler map was invalid.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause One possible cause is that the chassis scheduler map is applied to a specific interface. For most interface types, a scheduler map must be applied to all interfaces on the PIC;

therefore, a wildcard must be used to specify the interfaces. One exception to this rule is the Gigabit Ethernet IQ PIC.

Action Correct the configuration used to apply the chassis scheduler map to the interface.

COSD_CLASSIFIER_NO_SUPPORT_LSI

System Log Message Cannot support classifier type *classifier-type* on lsi interface *interface-name*

Description The Differentiated Services code point (DSCP) classifier and the 802.1p classifier are only supported on I-Chip based Flexible PIC Concentrators (FPCs).

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the DSCP or the 802.1p classifier configuration from the routing instance

COSD_CLASS_8021P_UNSUPPORTED

System Log Message ieee-802.1 classifier is not valid on interface *interface-name*

Description The IEEE 802.1p classifier is not supported on the indicated interface.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Remove the 802.1p classifier configuration from the interface, or configure an interface encapsulation type that supports 802.1p classifiers.

COSD_CLASS_NO_SUPPORT_IFD

System Log Message BA/Fixed Classifier or Rewrite on Physical Interface is not allowed when ethernet switching family is configured: interace *interface-name*

Description The Rewrite is not supported on this interface when ethernet switching is enabled

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the classifier configuration from the interface, instead apply it on the logical interface where ethernet switching family is enabled

Action Remove the Rewrite configuration from the interface, instead apply it on the logical interface where ethernet switching family is enabled

COSD_CLASS_NO_SUPPORT_L3_IFL

System Log Message BA/Fixed Classifier or Rewrite config is not allowed on logical interface (*interface-name*) with inet/inet6 family

Description The Rewrite is not supported on this logical interface

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the classifier configuration from the logical interface, instead apply it on the main interface if inet/inet6 is configured on one of its logical interfaces

Action Remove the Rewrite configuration from the logical interface, instead apply it on the main interface if inet/inet6 is configured on one of its logical interfaces

COSD_CONF_OPEN_FAILURE

System Log Message Unable to open: *filename*, using default CoS forwarding classes, do 'commit full' in cli to avoid this message

Description The class-of-service (CoS) process (cosd) could not read configuration data.

Type Error: An error occurred

Severity error

Facility ANY

Cause All of the following resons: mgd -I fails after upgrade-the file cosd.conf does not exist and is not created because of the mgd -I failure The first commit is 'commit' and not 'commit full'-the file cosd.conf does not commit and is not created automatically [class-of-service forwarding-classes] does not exist-the file cosd.conf does not get exported with plain 'commit'

Action Do a 'commit full'

COSD_DB_OPEN_FAILED

System Log Message Unable to open configuration database: *error-message*(*name*)

Description The class-of-service (CoS) process (cosd) could not read configuration data for the indicated reason.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause The specified database does not exist

Action Contact your technical support representative.

COSD_EXACT_RATE_UNSUPP_INTERFACE

System Log Message Unable to apply scheduler map *scheduler-map* to interface *interface-name* because it does not support exact-rate transmission

Description The class-of-service (CoS) process (cosd) did not apply the indicated scheduler map to the indicated interface, because a scheduler named in the scheduler map specifies exact transmission rate. The interface is housed on a type of PIC that does not support exact transmission rate, such as an IQ2 PIC. In terms of configuration, the 'exact' statement is included in the scheduler definition at the [edit class-of-service schedulers <scheduler-name> transmit-rate (<rate> | percent <percentage>)] hierarchy level. The scheduler is included in the scheduler map that is applied to the interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the 'exact' statement from the scheduler in the scheduler map applied to the interface.

COSD_EXACT_RATE_UNSUPP_SESSION

System Log Message Unable to apply CoS to L2TP session *session-id*, because scheduler map *scheduler-map* specifies exact rate transmission

Description The class-of-service (CoS) process (cosd) did not apply CoS settings to the indicated Layer 2 Tunneling Protocol (L2TP) session, because the scheduler map specified by the RADIUS server for the session is configured for exact transmission rate. Exact transmission rate is not supported for L2TP sessions on the type of PIC that houses the interface, such as an IQ2 PIC. In terms of configuration, the 'exact' statement is included in a scheduler definition at the [edit class-of-service schedulers <scheduler-name> transmit-rate (<rate> | percent <percentage>)] hierarchy level. The scheduler is included in a scheduler map that is associated with a traffic control profile. The traffic control profile is named by an attribute in the RADIUS server's configuration file, which makes the profile apply to the session.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the 'exact' statement from the scheduler in the scheduler map applied to the session.

COSD_EXP_RW_L2_IFL_NOT_SUPPORTED

System Log Message EXP Rewrite on IFL is not allowed when ethernet switching family is configured: interace *interface-name*

Description EXP rewrite is not supported on this logical interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the exp rewrite configuration from the logical interface.

COSD_FRAGMENTATION_MAP_CONFLICT

System Log Message Interface *compression-device* matches wildcard *wildcard-interface-name*, but fragmentation map *fragmentation-map* was not applied because interface is compression device for link interface *link-interface-name*

Description The indicated fragmentation map is normally applied to interfaces that match the indicated wildcard. The class-of-service (CoS) process (cosd) did not apply the fragmentation map to the indicated interface, even though it matches the wildcard, because the interface is acting as a compression device for the indicated link interface.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Correct the configuration of the fragmentation map.

COSD_HIGH_PRIO_QUEUES_INTERFACE

System Log Message Unable to apply scheduler map *scheduler-map* to interface *interface-name*, because multiple schedulers in map have "high,""medium-high," or "strict-high" priority

Description The class-of-service (CoS) process (cosd) did not apply the indicated scheduler map to the indicated interface, because the map includes more than one scheduler that has high, medium-high, or strict-high priority. For interfaces that are housed by certain PICs, such as an IQ2 PIC, the scheduler map can include only one scheduler that specifies one of those three priority levels. In terms of configuration, the 'priority' statement at the [edit class-of-service schedulers <scheduler-name>] hierarchy level has the value 'high, ' 'medium-high, ' or 'strict-high' for more than one of the schedulers in the map.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Correct the configuration so that the scheduler map includes only one scheduler with high, medium-high, or strict-high priority.

COSD_HIGH_PRIO_QUEUES_SESSION

System Log Message Unable to apply CoS to L2TP session *session-id*, because multiple schedulers in scheduler map *scheduler-map* have "high,""medium-high," or "strict-high" priority

Description The class-of-service (CoS) process (cosd) did not apply CoS settings to the indicated Layer 2 Tunneling Protocol (L2TP) session because the scheduler map specified by the RADIUS server for the session includes more than one scheduler that has high, medium-high, or strict-high priority. For interfaces that are housed by certain Physical Interface Cards (PICs), such as an IQ2 PIC, the scheduler map can include only one scheduler that specifies one of those three priority levels. In terms of configuration, the 'priority' statement at the [edit class-of-service schedulers <scheduler-name>] hierarchy level has the value 'high, ' 'medium-high, ' or 'strict-high' for more than one of the schedulers in the map. The map is associated with a traffic control profile that is named by an attribute in the RADIUS server's configuration file, which makes the profile apply to the session.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Correct the configuration so that the scheduler map includes only one scheduler with high, medium-high, or strict-high priority.

COSD_IFD_OUTPUT_SHAPING_RATE_ERR

System Log Message Traffic shaping not supported on interface device *interface-name*

Description The class-of-service (CoS) process (cosd) did not apply the shaping rate that is configured for the indicated interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause Shaping rate is valid only for interfaces housed by IQ and IQ2 PICs, and the interface is on a different type of PIC.

Action Remove the shaping rate configuration from the interface.

COSD_IFD_SHAPER_ERR

System Log Message port shaper not allowed on interface *interface-name*

Description The non-queuing dense port concentrators (DPCs) did not support the specified shaping rate.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause The port shaper was not supported on the non-queuing DPCs.

Action Remove the shaping rate configuration from the interface.

COSD_INTERFACE_NO_MEDIA

System Log Message Unable to obtain media information for interface *interface-name*

Description The message sent by the kernel for the indicated interface did not include required media information.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause An internal software failure occurred.

Action Contact your technical support representative.

COSD_L2TP_COS_NOT_CONFIGURED

System Log Message Unable to apply CoS to L2TP session *session-id* because session-aware CoS is not enabled for interface *interface-name*

Description The class-of-service (CoS) process (cosd) did not apply CoS settings to the indicated Layer 2 Tunneling Protocol (L2TP) session on the indicated interface, because the interface is not configured to support session-aware CoS for L2TP. In terms of configuration, the 'per-session-scheduler' statement is not included at the [edit interfaces <interface-name> unit <logical-unit-number>] hierarchy level.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Include the 'per-session-scheduler' statement in the configuration for the interface.

COSD_L2TP_COS_NOT_SUPPORTED

System Log Message Unable to apply CoS to L2TP session *session-id* on interface *interface-name*: it does not support CoS

Description The class-of-service (CoS) process (cosd) did not apply CoS settings to the indicated Layer 2 Tunneling Protocol (L2TP) session on the indicated interface. The interface is configured to support session-aware CoS for L2TP, but is not on a PIC that supports that feature, such as an IQ2 PIC. In terms of configuration, the 'per-session-scheduler'

statement is included at the [edit interfaces <interface-name> unit <logical-unit-number>] hierarchy level.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Determine whether the interface is on an PIC that supports session-aware CoS; if not, remove the 'per-session-scheduler' statement.

COSD_L2TP_SHAPING_NOT_CONFIGURED

System Log Message Unable to apply CoS to L2TP session *session-id* because session-aware shaping is not enabled for interface *interface-name*

Description The class-of-service (CoS) process (cosd) did not apply CoS settings to the indicated Layer 2 Tunneling Protocol (L2TP) session on the indicated interface, because session-aware traffic shaping for L2TP is not configured on the PIC that houses the interface. In terms of configuration, the 'session-shaping' statement is not included at the [edit chassis fpc <slot-number> pic <pic-number> traffic-manager mode] hierarchy level.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Include the 'session-shaping' statement in the configuration for the PIC.

COSD_LARGE_DELAY_BUFFER_INVALID

System Log Message Error for interface *interface-name error-message*

Description The class-of-service (CoS) process (cosd) did not apply the large delay buffer setting that is configured for the indicated interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause The interface is not housed on one of the PIC types that support large delay buffer.

Action Remove the large delay buffer configuration from the interface.

COSD_MALLOC_FAILED

System Log Message malloc failed: *error-message*

Description The class-of-service (CoS) process (cosd) could not dynamically allocate memory for the indicated reason.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause A software bug caused a memory leak or the Routing Engine did not have sufficient memory.

Action Contact your technical support representative. For more information, see http://kb.juniper.net/InfoCenter/index?page=content&id=KB18862.

COSD_MAX_FORWARDING_CLASSES_ABC

System Log Message exceeding max 4 forwarding-class support.

Description User configuration exceeds the maximum number of forwarding class that is supported.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Configure only four forwarding classes

COSD_MPLS_DSCP_CLASS_NO_SUPPORT

System Log Message Cannot support MPLS DSCP classifier on ifl *interface-name*

Description The MPLS Differentiated Services code point (DSCP) classifier is only supported on I-Chip based Flexible PIC Concentrators (FPCs). It is not supported on Q2 PICs.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the MPLS DSCP classifier configuration from the logical interface.

COSD_MULTILINK_CLASS_CONFLICT

System Log Message Fragmentation map *fragmentation-map* for wildcard *wildcard-interface-name* specified multilink class *class-name* for queue *queue-number* on interface *interface-name*, which exceeds configured limit of *limit*

Description The indicated fragmentation map is normally applied to interfaces that match the indicated wildcard, and specifies the indicated multilink class setting for queues on those interfaces. The class-of-service (CoS) process (cosd) did not apply the fragmentation

map to the indicated interface, even though it matches the wildcard, because the setting in the map exceeds the indicated class limit, which is configured on the interface itself.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Correct the configuration so that the multilink class setting in the fragmentation map does not exceed the class limit for the interface.

COSD_NULL_INPUT_ARGUMENT

System Log Message NULL input argument : *error-message*

Description The pointer that was passed to this function was NULL.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Contact your technical support representative.

COSD_OUT_OF_DEDICATED_QUEUES

System Log Message Queue usage count for interface *interface-name* is at *percentage-value* percent

Description The class-of-service (CoS) process (cosd) is running out of dedicated queues.

Type Event: This message reports an event, not an error

Severity warning

Facility LOG_DAEMON

COSD_RATE_LIMIT_INVALID

System Log Message Unable to apply scheduler map *scheduler-map* to interface *interface-name*. *description*.

Description The class-of-service (CoS) process (cosd) did not apply the indicated scheduler map to the indicated interface, because the number of rate limited queues in the scheduler map exceeded the limit supported by this interface or the priority is not supported. The interface is housed in a type of PIC that does not support the number of configured rate limited queues or the priority is not supported. In terms of configuration, the 'rate-limit' statement is included in the scheduler definition at the [edit class-of-service schedulers <scheduler-name> transmit-rate <rate> | percent <percentage>] hierarchy level. The scheduler is included in the scheduler map applied to the interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Either limit the number of rate-limited schedulers in this scheduler map to the allowed maximum for this PIC and interface type or check the allowed priority for rate-limited queues

COSD_RATE_LIMIT_NOT_SUPPORTED

System Log Message Unable to apply scheduler map *scheduler-map* to interface *interface-name* because it does not support rate limiting

Description The class-of-service (CoS) process (cosd) did not apply the indicated scheduler map to the indicated interface, because a scheduler named in the scheduler map is configured for rate limiting. The interface is housed in a type of PIC that does not support rate limiting. In terms of configuration, the 'rate-limit' statement is included in the scheduler definition at the [edit class-of-service schedulers <scheduler-name> transmit-rate <rate> | percent <percentage>] hierarchy level. The scheduler is included in the scheduler map applied to the interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove the 'rate-limit' statement from the scheduler in the scheduler map applied to the interface.

COSD_REWRITE_RULE_LIMIT_EXCEEDED

System Log Message Number of rewrite rules applied to interface *interface-name* exceeds limit (*maximum-value*)

Description The class-of-service (CoS) process (cosd) determined that the number of rewrite rules applied to the indicated interface exceeds the indicated limit for the interface. In terms of configuration, too many rewrite rules are included at the [edit class-of-service interfaces <interface-name> unit <logical-unit-number> rewrite-rules] hierarchy level.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Remove rewrite rules from the configuration for the interface.

COSD_RL_IFL_NEEDS_SHAPING

System Log Message "rate-limit" configured in *scheduler-map*, but ifl *interface-name* does not have output shaper configured. It will use the ifd-shaping rate/ifd-transmit rate for implementation of rate-limit.

Description The 'rate-limit' statement is configured in one or more schedulers that are part of the indicated scheduler map. In order to apply this scheduler map to the indicated interface,

output shaping rate should be configured on the interface. Since no output shaping rate is configured, the transmit rate or shaping rate of the parent interface will be used instead.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Configure output shaping rate for the indicataed interface

COSD_SCHEDULER_MAP_CONFLICT

System Log Message Forwarding classes "*first-forwarding-class*" and "*second-forwarding-class*" in scheduler map *scheduler-map* both map to queue *queue-number*

Description Both of the indicated forwarding classes, which are defined in the indicated scheduler map, map to the same indicated queue. The double mapping is invalid.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Map only one forwarding class to the queue.

COSD_SCHED_AVG_CONST_UNSUPPORTED

System Log Message Averaging constant not supported on interface *interface-name*. Value set in scheduler-map *scheduler-map* (scheduler *name*) will be ignored.

Description Configuring averaging constant is not supported on the indicated interface. Value set in the indicated scheduler will be ignored.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Remove the averaging-constant configuration from the indicated scheduler.

COSD_SCHED_MAP_GROUP_CONFLICT

System Log Message Interface *interface-name* cannot be bound to scheduler-map *scheduler-map*. It will be bound to default scheduler-map

Description Interfaces belonging to a group cannot be bound to different scheduler maps. They will be bound to the default scheduler map.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Map only one scheduler map to all the interfaces of a group.

COSD_SHAPER_GROUP_CONFLICT

System Log Message Interface *interface-name* cannot be bound to configured shaping-rate. It will be bound to default rate

Description Interfaces belonging to a group cannot be bound to different shaping rates. They will be bound to the default shaping rate.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Map only one shaping rate to all interfaces of a group.

COSD_STREAM_IFD_CREATE_FAILURE

System Log Message Unable to create special master interface device for *interface-name*

Description The class-of-service (CoS) process (cosd) could not create the indicated internal interface device, which it needs for application of a chassis scheduler map.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause An internal software failure occurred.

Action Contact your technical support representative.

COSD_TIMER_ERROR

System Log Message Unable to set retry timer for rtsock write operation: *error-message*

Description The class-of-service (CoS) process (cosd) used a routine from the rtsock library to write to the kernel, but the kernel did not accept the request. The cosd process could not set the retry timer for the request, for the indicated reason.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Cause An internal software failure occurred.

Action Contact your technical support representative.

COSD_TRICOLOR_ALWAYS_ON

System Log Message tri-color is always enabled in this platform. There is no need to explicitly set it.

Description Tri-color marking is always enabled on this platform. There is no need to explicitly set it.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Action Remove the 'tri-color' configuration statement

COSD_TRICOLOR_NOT_SUPPORTED

System Log Message Unable to apply scheduler *scheduler-map* to interface *interface-name*, because it does not support tricolor marking

Description The class-of-service (CoS) process (cosd) did not apply the indicated scheduler map to the indicated interface, because a scheduler included in the map specifies a packet loss priority (PLP) that is supported only with tricolor marking (TCM). The interface does not support TCM, either because TCM is not enabled or the interface is on a router that does not support TCM. In terms of configuration, the value 'medium-high' or 'medium-low' is specified for the 'loss-priority' statement in a scheduler definition at the [edit class-of-service schedulers <scheduler-name> drop-profile-map] hierarchy level. The scheduler is included in the scheduler map applied to the interface, but the 'tri-color' statement is either not included at the [edit class-of-service] hierarchy level, or is not supported.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Change the value of the 'loss-priority' statement in the scheduler or include the 'tri-color' statement to enable TCM on the router.

COSD_TX_QUEUE_RATES_TOO_HIGH

System Log Message Unable to apply scheduler map *scheduler-map* to interface *interface-name*: sum of scheduler transmission rates exceeds interface shaping or transmission rate

Description The class-of-service (CoS) process (cosd) did not apply the indicated scheduler map to the indicated interface, because the sum of the queue transmission rates defined in the schedulers in the scheduler map exceeds the shaping or transmission rate for the interface. In terms of configuration, the 'transmit-rate' statement is specified for each scheduler at the [edit class-of-service schedulers <scheduler-name>] hierarchy level. The sum of the configured transmission rates exceeds the transmission or shaping rate of the interface.

Type Error: An error occurred

Severity error

Facility LOG_DAEMON

Action Decrease the value of one or more 'transmit-rate' statements so that the sum is less than the interface transmission or shaping rate.

COSD_UNKNOWN_CLASSIFIER

System Log Message classifier type *classifier-type* is invalid

Description The class-of-service (CoS) process (cosd) did not recognize the indicated classifier type from the rtsock library.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Cause An internal software failure occurred.

Action Contact your technical support representative.

COSD_UNKNOWN_REWRITE

System Log Message rtsock rewrite type *type* is invalid

Description The class-of-service (CoS) process (cosd) did not recognize the indicated rewrite type from the rtsock library.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Cause An internal software failure occurred.

Action Contact your technical support representative.

COSD_UNKNOWN_TRANSLATION_TABLE

System Log Message rtsock translation table type *translation-table-type* is invalid

Description The class-of-service (CoS) process (cosd) did not recognize the indicated translation table type from the rtsock library.

Type Error: An error occurred

Severity warning

Facility LOG_DAEMON

Cause An internal software failure occurred.

Action For more information, see http://kb.juniper.net/InfoCenter/index?page=content&id=KB18866.

Collecting Subscriber Access Logs Before Contacting Juniper Technical Support

Problem When you experience a subscriber access problem in your network, we recommend that you collect certain logs before you contact Juniper Technical Support. This topic shows you the most useful logs for a variety of network implementations. In addition to the relevant log information, you must also collect standard troubleshooting information and send it to Juniper Technical Support in your request for assistance.

Solution To collect standard troubleshooting information:

- Redirect the command output to a file.

  user@host> **request support information | save rsi-1**

To configure logging to assist Juniper Technical Support:

1. Review the following blocks of statements to determine which apply to your configuration.

   ```
   [edit]
   set system syslog archive size 100m files 25
   set system auto-configuration traceoptions file filename
   set system auto-configuration traceoptions file filename size 100m files 25
   set protocols ppp-service traceoptions file filename size 100m files 25
   set protocols ppp-service traceoptions level all
   set protocols ppp-service traceoptions flag all
   set protocols ppp traceoptions file filename size 100m files 25
   set protocols ppp traceoptions level all
   set protocols ppp traceoptions flag all
   set protocols ppp monitor-session all
   set interfaces pp0 traceoptions flag all
   set demux traceoptions file filename size 100m files 25
   set demux traceoptions level all
   set demux traceoptions flag all
   set system processes dhcp-service traceoptions file filename
   set system processes dhcp-service traceoptions file size 100m
   set system processes dhcp-service traceoptions file files 25
   set system processes dhcp-service traceoptions flag all
   set class-of-service traceoptions file filename
   set class-of-service traceoptions file size 100m
   set class-of-service traceoptions flag all
   set class-of-service traceoptions file files 25
   set routing-options traceoptions file filename
   set routing-options traceoptions file size 100m
   set routing-options traceoptions flag all
   set routing-options traceoptions file files 25
   set interfaces traceoptions file filename
   set interfaces traceoptions file size 100m
   set interfaces traceoptions flag all
   set interfaces traceoptions file files 25
   set system processes general-authentication-service traceoptions file filename
   set system processes general-authentication-service traceoptions file size 100m
   set system processes general-authentication-service traceoptions flag all
   set system processes general-authentication-service traceoptions file files 25
   ```

2. Copy the relevant statements into a text file and modify the log filenames as you want.
3. Copy the statements from the text file and paste them into the CLI on your router to configure logging.
4. Commit the logging configuration to begin collecting information.

NOTE: The maximum file size for DHCP local server and DHCP relay log files is 1 GB. The maximum number of log files for DHCP local server and DHCP relay is 1000.

BEST PRACTICE: Enable these logs only to collect information when troubleshooting specific problems. Enabling these logs during normal operations can result in reduced system performance.

Related Documentation

- *Compressing Troubleshooting Logs from /var/logs to Send to Juniper Technical Support*

PART 5

Index

Index

Symbols

A

B

C

T

U

V